How to Start a Startup
Everything we know about how to start a startup, for free, from some of the world experts.

http://startupclass.samaltman.com/

CS183B is a class we’re teaching at Stanford. It’s designed to be a sort of one-class business course for people who want to start startups.

Videos of the lectures, associated reading materials, and assignments will all be available here. There will be 20 videos, some with a speaker or two and some with a small panel. It’ll be 1,000 minutes of content if you watch it all.

We’ll cover how to come up with ideas and evaluate them, how to get users and grow, how to do sales and marketing, how to hire, how to raise money, company culture, operations and management, business strategy, and more.

You can’t teach everything necessary to succeed in starting a company, but I suspect we can teach a surprising amount. We’ve tried to take some of the best speakers from the past 9 years of Y Combinator dinners and arrange them in a way that will hopefully make sense.

We’re doing this because we believe helping a lot of people be better at starting companies will be good for everyone. It will hopefully be valuable even for people who don’t want to start startups. Talks like these have really helped Y Combinator founders create their companies. We hope you find it helpful too!

-Sam

Following along

All lecture videos will be uploaded to this site - at 4 PM Pacific Time, after the in-person lectures every Tuesday and Thursday. Hundreds of universities are organizing groups to watch the 50-minute videos together, as well as peer evaluate the projects. See the list of groups and viewing sessions at universities and other organizations. If there isn’t a viewing session at your university or being hosted at a local organization, you can apply to be a leader.

In addition to the mailing list and Facebook group, a discussion forum to discuss the contents of each lecture and reading is here.
1. Ideas, products, teams and execution I
   Lecture 1: Advice for Ambitious 19 Year Olds
   Lecture 1: Good and Bad Reasons to Become an Entrepreneur

2. Ideas, products, teams and execution II
   Lecture 2: Stupid Apps and Changing the World
   Lecture 2: Do Things That Don't Scale

3. How to have Ideas
   Lecture 3: How to Get Startup Ideas
   Lecture 3: 1995 Interview with Computerworld's Oral History Project (Steve Jobs)

4. Building product, talking and growing
   Lecture 4: From failure to $37M funding with Adora Cheung – Homejoy
   Lecture 4: Why Startups Need to Focus on Sales, Not Marketing

5. Business strategy and monopoly
   Lecture 5: Competition Is for Losers
   Lecture 5: Ben Horowitz to Startup CEOs: This Won't Be Easy

6. Growth
   Lecture 6: How we put Facebook on the path to 1 billion users
   Lecture 6: Growth Accounting & Triangle Heatmap Explanation
   Lecture 6: Does Slow Growth Equal Slow Death?
   Lecture 6: A Recipe for Growth: Adding Layers to the Cake
   Lecture 6: Mark Zuckerberg on Facebook's Early Days: Go Hard or Go Home
   Lecture 6: Startup = growth
   Lecture 6: Chasing Facebook's Next Billion Users
   Lecture 6: The only metric that matters

7. How to build products users love
   Lecture 7: Your app makes me fat
   Lecture 7: What Makes a Design Seem 'Intuitive'?
<table>
<thead>
<tr>
<th>Lecture</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Creative mornings with Ben Chestnut [video transcript]</td>
<td>148</td>
</tr>
<tr>
<td>7</td>
<td>What Makes Marriage Work?</td>
<td>154</td>
</tr>
<tr>
<td>7</td>
<td>(BONUS) BJ Fogg's Behavior Model</td>
<td>158</td>
</tr>
<tr>
<td>7</td>
<td>(BONUS): Customer Intimacy and Other Value Disciplines</td>
<td>159</td>
</tr>
<tr>
<td>7</td>
<td>(BONUS) The sanctity of Marriage</td>
<td>160</td>
</tr>
<tr>
<td>7</td>
<td>(BONUS) Creating Passionate Users</td>
<td>160</td>
</tr>
<tr>
<td>8</td>
<td>Doing things that don’t scale</td>
<td>161</td>
</tr>
<tr>
<td>8</td>
<td>The Press is a Tool</td>
<td>174</td>
</tr>
<tr>
<td>9</td>
<td>How to raise money</td>
<td>177</td>
</tr>
<tr>
<td>9</td>
<td>Why Software Is Eating The World</td>
<td>190</td>
</tr>
<tr>
<td>9</td>
<td>How to Convince Investors</td>
<td>191</td>
</tr>
<tr>
<td>9</td>
<td>How to raise money</td>
<td>195</td>
</tr>
<tr>
<td>9</td>
<td>Linkedin’s Series B Picture to Greylock by Reid Hoffman</td>
<td>206</td>
</tr>
<tr>
<td>9</td>
<td>Ron Conway’s Recommended Reading List</td>
<td>223</td>
</tr>
<tr>
<td>10</td>
<td>Company culture &amp; building a team I</td>
<td>226</td>
</tr>
<tr>
<td>10</td>
<td>The Happiness Culture: Zappos isn’t a company -it’s a mission</td>
<td>235</td>
</tr>
<tr>
<td>10</td>
<td>Don’t Fuck Up the Culture</td>
<td>237</td>
</tr>
<tr>
<td>11</td>
<td>Company culture &amp; building a team II</td>
<td>239</td>
</tr>
<tr>
<td>11</td>
<td>What it’s like to work for Stripe</td>
<td>251</td>
</tr>
<tr>
<td>11</td>
<td>How to hire</td>
<td>253</td>
</tr>
<tr>
<td>12</td>
<td>Building for the enterprise</td>
<td>257</td>
</tr>
<tr>
<td>12</td>
<td>The Continuous Productivity of Aaron Levy</td>
<td>268</td>
</tr>
<tr>
<td>12</td>
<td>Robert Cialdini</td>
<td>272</td>
</tr>
<tr>
<td>12</td>
<td>Marc Andreessen On The Future Of Enterprise</td>
<td>273</td>
</tr>
<tr>
<td>13</td>
<td>Being a great founder</td>
<td>282</td>
</tr>
<tr>
<td>13</td>
<td>The Information Age to the Networked Age: Are You Network Literate?</td>
<td>288</td>
</tr>
<tr>
<td>13</td>
<td>The Alliance: A Visual Summary</td>
<td>290</td>
</tr>
<tr>
<td>13</td>
<td>What I Wish I Knew Before Pitching LinkedIn to VCs</td>
<td>290</td>
</tr>
<tr>
<td>13</td>
<td>If, Why, and How Founders Should Hire a “Professional” CEO</td>
<td>292</td>
</tr>
<tr>
<td>13</td>
<td>Want to Start a Startup?</td>
<td>296</td>
</tr>
</tbody>
</table>

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208) 4 / 412
14. How to operate

Lecture 14: Readings

15. How to manage

Lecture 15: Making Yourself a CEO
Lecture 15: A Good Place to Work
Lecture 15: How to Minimize Politics in Your Company

16. How to run a user interview

17. How to design hardware products

Lecture 17: Hosain Rahman at Startup School SV 2014

18. Mechanics: legal, finance, hr...

Lecture 18: How to Work with Lawyers at a Startup
Lecture 18: Useful resources
Lecture 18: Top 10 reasons why entrepreneurs hate lawyers
Lecture 18: Startup Company Lawyer

19. How to pitch

Lecture 19: Five ways to build a $100 million business
Lecture 19: SaaStr – From-the-trenches sales advice
Lecture 19: How I Raised Myself From Failure to Success In Selling
Lecture 19: Pitching Hacks! How to pitch startups to investors

20. Closing thoughts
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>23 sept <strong>Welcome, and Ideas, Products, Teams and Execution Part I</strong></td>
<td><strong>Sam Altman, President Y Combinator</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Why to Start a Startup</strong></td>
<td><strong>Dustin Moskovitz, Cofounder Facebook, Asana, Good Ventures</strong></td>
</tr>
<tr>
<td>2</td>
<td>25 sept <strong>Ideas, Products, Teams and Execution Part II</strong></td>
<td><strong>Sam Altman, President Y Combinator</strong></td>
</tr>
<tr>
<td>3</td>
<td>30 sept <strong>Counterintuitive Parts of Startups, and How to Have Ideas</strong></td>
<td><strong>Paul Graham, Founder Y Combinator</strong></td>
</tr>
<tr>
<td>4</td>
<td>2 okt. <strong>Building Product, Talking to Users, and Growing</strong></td>
<td><strong>Adora Cheung, Founder Homejoy</strong></td>
</tr>
<tr>
<td>5</td>
<td>7 okt. <strong>Business Strategy and Monopoly Theory</strong></td>
<td><strong>Peter Thiel, Founder Paypal</strong></td>
</tr>
<tr>
<td>6</td>
<td>9 okt. <strong>Growth</strong></td>
<td><strong>Alex Schultz, VP Growth Facebook</strong></td>
</tr>
<tr>
<td>7</td>
<td>14 okt. <strong>How to Build Products Users Love</strong></td>
<td><strong>Kevin Hale, Founder Wufoo</strong></td>
</tr>
<tr>
<td>8</td>
<td>16 okt. <strong>Doing Things That Don’t Scale PR</strong></td>
<td><strong>Walker Williams, Founder Teespring</strong></td>
</tr>
<tr>
<td></td>
<td><strong>How to Get Started</strong></td>
<td><strong>Justin Kan, Founder Twitch TV</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Stanley Tang, Founder DoorDash</strong></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>21 okt. <strong>How to Raise Money</strong></td>
<td><strong>Marc Andreessen, Founder Andreessen Horowitz, Netscape</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Ron Conway, Founder SV Angel</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Parker Conrad, Founder Zenefits</strong></td>
</tr>
<tr>
<td>10</td>
<td>23 okt. <strong>Company Culture and Building a Team, Part I</strong></td>
<td><strong>Alfred Lin, Former COO Zappos</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Brian Chesky, Founder Airbnb</strong></td>
</tr>
<tr>
<td>11</td>
<td>28 okt. <strong>Company Culture and Building a Team, Part II</strong></td>
<td><strong>Patrick Collison, Founders Stripe</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>John Collison, Founders Stripe</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Ben Silbermann, Founder &amp; CEO Pinterest</strong></td>
</tr>
<tr>
<td>12</td>
<td>30 okt. <strong>Sales and Marketing</strong></td>
<td><strong>Aaron Levie, Founder Box</strong></td>
</tr>
<tr>
<td>13</td>
<td>4 nov. <strong>How To Be A Great Founder</strong></td>
<td><strong>Reid Hoffman, Founder LinkedIn</strong></td>
</tr>
<tr>
<td>14</td>
<td>6 nov. <strong>How to Operate</strong></td>
<td><strong>Keith Rabois, Partner Khosla Ventures</strong></td>
</tr>
<tr>
<td>15</td>
<td>11 nov. <strong>How to Manage</strong></td>
<td><strong>Ben Horowitz, Founder Andreessen Horowitz, and Opsware</strong></td>
</tr>
<tr>
<td>16</td>
<td>13 nov. <strong>How to Run a User Interview</strong></td>
<td><strong>Emmett Shear, Founder and CEO Twitch</strong></td>
</tr>
<tr>
<td>17</td>
<td>18 nov. <strong>How to Design Hardware Products</strong></td>
<td><strong>Hosain Rahman, Founder Jawbone</strong></td>
</tr>
<tr>
<td>18</td>
<td>20 nov. <strong>Mechanics: Legal, Finance, HR, etc.</strong></td>
<td><strong>Kirsty Nathoo, Partner Y Combinator</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Carolynn Levy, Partner Y Combinator</strong></td>
</tr>
<tr>
<td>19</td>
<td>2 dec. <strong>Sales and Marketing</strong></td>
<td><strong>Tyler Bosmeny, Founder and CEO Clever</strong></td>
</tr>
<tr>
<td></td>
<td><strong>How To Pitch</strong></td>
<td><strong>Michael Seibel, Partner Y Combinator</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Investor Meeting Roleplaying</strong></td>
<td><strong>Qasar Younis, Partner Y Combinator</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Dalton Caldwell, Partner Y Combinator</strong></td>
</tr>
<tr>
<td>20</td>
<td>4 dec. <strong>Closing Thoughts and Later-Stage Advice</strong></td>
<td><strong>Sam Altman, President Y Combinator</strong></td>
</tr>
</tbody>
</table>

[http://startupclass.samaltman.com/] (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Projects

Projects will be peer evaluated, on the HTSAS Projects subreddit. Please see the rules on the subreddit regarding how to correctly format your submission.
There is no set due date - you are free to submit your projects to the subreddit at any time.

**Project 0**, released 9/23/14

For two weeks, keep a log of all the things that seem to be missing and all the problems you encounter in life.

**Notes:**
Use a platform that allows you to easily add to the log on your phone. You want to be able to pull out your phone and jot something down quickly whenever you notice something missing or problematic. This reading is helpful for context: [www.paulgraham.com/startupideas.html](http://www.paulgraham.com/startupideas.html) As PG suggests, turn off all of your other filters, including whether the thing that’s missing could be a startup idea. Also don’t worry about whether the problems you encounter are too trivial. The idea is to get this background process running, and get you in the habit of noticing more potential startup ideas.
By the end of the 2 weeks, there should be plenty of items in your log.

**Project 1**, released 9/30/14

Come up with 5 business ideas. For each, briefly answer the following questions: what are you going to build? Who needs it? How do you know? Why is this a good idea?

**Note:**
The ideas don’t need to be related to any of the things in your log, but it would be surprising if none of those problems had a related business idea.

**Project 2**, released 10/7/14

Choose one of your ideas from Project 1, and refine it. Write a 2 page overview fleshing out answers to the questions:

- What are you going to build?
  - What are the primary features? What will it generally look like?
- Who needs it?
  - Why? What are they using right now?
- Why is this a good idea?
- Who are your competitors?
- How will you get users?

If you can hack, between now and Project 3 - get feedback from users, you should build out a prototype; no matter how simple or rough, you will find user feedback to be much more meaningful if you have something they can play with.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
1. Ideas, products, teams and execution I
Lecture 1: Welcome

http://startupclass.samaltman.com/courses/lec01/

Sam Altman

Welcome to CS183B. I am Sam Altman, I'm the President of Y Combinator. Nine years ago, I was a Stanford student, and then I dropped out to start a company and then I've been an investor for the last few. So YC, we've been teaching people how to start startups for nine years. Most of it's pretty specific to the startups but thirty percent of it is pretty generally applicable. And so we think we can teach that thirty percent in this class. And even though that's only thirty percent of the way there, hopefully it will still be really helpful.

We've taught a lot of this class at YC and it's all been off the record. And this is the first time a lot of what we teach is going to be on the record. We've invited some of our guest speakers to come and give the same talks they give at YC. We've now funded 725 companies and so we're pretty sure a lot of this advice we give is pretty good. We can't fund every startup yet, but we can hopefully make this advice very generally available.

I'm only teaching three. Counting YC itself, every guest speaker has been involved in the creation of a billion plus dollar company. So the advice shouldn't be that theoretical, it's all been people who have done it.

All of the advice in this class is geared towards people starting a business where the goal is hyper growth and eventually building a very large company. Much of it doesn't apply in other cases and I want to warn people up front, that if you try to do these things in a lot of big companies or non-startups, it won't work. It should still be interesting, I really think that startups are the way of the future and it's worth trying to understand them, but startups are very different than normal companies. So over the course of today and Thursday, I'm going to try to give an overview of the four areas you need to excel at in order to maximize your success as a startup. And then throughout the course, the guest speakers are going to drill into all of these in more detail.
Lecture 1: Ideas, Products, Teams and Execution Part I

So the four areas: You need a great idea, a great product, a great team, and great execution. These overlap somewhat, but I'm going to have to talk about them somewhat individually to make it make sense.

You may still fail. The outcome is something like idea x product x execution x team x luck, where luck is a random number between zero and ten thousand. Literally that much. But if you do really well in the four areas you can control, you have a good chance at at least some amount of success.

One of the exciting things about startups is that they are a surprisingly even playing field. Young and inexperienced, you can do this. Old and experienced, you can do this, too. And one of the things that I particularly like about startups is that some of the things that are bad in other work situations, like being poor and unknown, are actually huge assets when it comes to starting a startup.

Before we jump in on the how, I want to talk about why you should start a startup. I'm somewhat hesitant to be doing this class at all because you should never start a startup just for the sake of doing so. There are much easier ways to become rich and everyone who starts a startup always says, always, that they couldn't have imagined how hard and painful it was going to be. You should only start a startup if you feel compelled by a particular problem and that you think starting a company is the best way to solve it.

The specific passion should come first, and the startup second. In fact, all of the classes we have at YC follow this. So for the second half of today's lecture, Dustin Moskovitz is going to take over and talk about why to start a startup. We were so surprised at the amount of attention this class got, that we wanted to make sure we spent a lot of time on the why.

The first of the four areas: a great idea. It's become popular in recent years to say that the idea doesn't matter. In fact, it's uncool to spend a lot of time thinking about the idea for a startup. You're just supposed to start, throw stuff at the wall, see what sticks, and not even spend any time thinking about if it will be valuable if it works.

And pivots are supposed to be great, the more pivots the better. So this isn't totally wrong, things do evolve in ways you can't totally predict. And there's a limit to how much you can figure out without actually getting a product in the hands of the users. And great execution is at least ten times as important and a hundred times harder than a great idea.

But the pendulum has swung way out of whack. A bad idea is still bad and the pivot-happy world we're in today feels suboptimal. Great execution towards a terrible idea will get you nowhere. There are exceptions, of course, but most great companies start with a great idea, not a pivot.

If you look at successful pivots, they almost always are a pivot into something the founders themselves wanted, not a random made up idea. Airbnb happened because Brian Chesky couldn't pay his rent, but he had some extra space. In general though if you look at the track record of pivots, they don't become big companies. I myself used to believe ideas didn't matter that much, but I'm very sure that's wrong now.
The definition of the idea, as we talk about it, is very broad. It includes the size and the growth of the market, the growth strategy for the company, the defensibility strategy, and so on. When you're evaluating an idea, you need to think through all these things, not just the product. If it works out, you're going to be working on this for ten years so it's worth some real up front time to think through the up front value and the defensibility of the business. Even though plans themselves are worthless, the exercise of planning is really valuable and totally missing in most startups today.

Long-term thinking is so rare anywhere, but especially in startups. There is a huge advantage if you do it. Remember that the idea will expand and become more ambitious as you go. You certainly don't need to have everything figured out in your path to world domination, but you really want a nice kernel to start with. You want something that can develop in interesting ways.

As you're thinking through ideas, another thing we see that founders get wrong all the time is that someday you need to build a business that is difficult to replicate. This is an important part of a good idea.

I want to make this point again because it is so important: the idea should come first and the startup should come second. Wait to start a startup until you come up with an idea you feel compelled to explore. This is also the way to choose between ideas. If you have several ideas, work on the one that you think about most often when you're not trying to think about work. What we hear again and again from founders is that they wish they had waited until they came up with an idea they really loved. Another way of looking at this is that the best companies are almost always mission oriented. It's difficult to get the amount of focus that large companies need unless the company feels like it has an important mission. And it's usually really hard to get that without a great founding idea. A related advantage of mission oriented ideas is that you yourself will be dedicated to them. It takes years and years, usually a decade, to build a great startup. If you don't love and believe in what you're building, you're likely to give up at some point along the way. There's no way I know of to get through the pain of a startup without the belief that the mission really matters. A lot of founders, especially students, believe that their startups will only take two to three years and then after that they'll work on what they're really passionate about. That almost never works. Good startups usually take ten years.

A third advantage of mission oriented companies is that people outside the company are more willing to help you. You'll get more support on a hard, important project, than a derivative one. When it comes to starting a startup, it's easier to found a hard startup than an easy startup. This is one of those counter-intuitive things that takes people a long time to understand. It's difficult to overstate how important being mission driven is, so I want to state it one last time: derivative companies, companies that copy an existing idea with very few new insights, don't excite people and they don't compel the teams to work hard enough to be successful.

Paul Graham is going to talk about how to get startup ideas next week. It's something that a lot of founders struggle with, but it's something I believe you can get better at with practice and it's definitely worth trying to get better at.

The hardest part about coming up with great ideas, is that the best ideas often look terrible at the beginning. The thirteenth search engine, and without all the features of a web portal? Most people thought that was pointless. Search was done, and anyways, it didn't matter that much. Portals were where the value was at. The tenth social network, and limited only to college students with no money? Also terrible. MySpace has won and who wants college students as customers? Or a way to stay on strangers' couches. That just sounds terrible all around.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
These all sounded really bad but they turned out to be good. If they sounded really good, there would be too many people working on them. As Peter Thiel is going to discuss in the fifth class, you want an idea that turns into a monopoly. But you can't get a monopoly right away. You have to find a small market in which you can get a monopoly and then quickly expand. This is why some great startup ideas look really bad at the beginning. It's good if you can say something like, "Today, only the small substantive users are going to use my product, but I'm going to get all of them, and in the future, almost everyone is going to use my product."

Here is the theme that is going to come up a lot: you need conviction in your own beliefs and a willingness to ignore others' naysaying. The hard part is that this is a very fine line. There's right on one side of it, and crazy on the other. But keep in mind that if you do come up with a great idea, most people are going to think it's bad. You should be happy about that, it means they won't compete with you.

That's why it's also not dangerous to tell people your idea. The truly good ideas don't sound like they're worth stealing. You want an idea where you can say, "I know it sounds like a bad idea, but here's specifically why it's actually a great one." You want to sound crazy, but you want to actually be right. And you want an idea that not many other people are working on. And it's okay if it doesn't sound big at first.

A common mistake among founders, especially first time founders, is that they think the first version of their product - the first version of their idea - needs to sound really big. But it doesn't. It needs to take over a small specific market and expand from there. That's how most great companies get started. Unpopular but right is what you're going for. You want something that sounds like a bad idea, but is a good idea.

You also really want to take the time to think about how the market is going to evolve. You need a market that's going to be big in 10 years. Most investors are obsessed with the market size today, and they don't think at all about how the market is going to evolve.

In fact, I think this is one of the biggest systemic mistakes that investors make. They think about the growth of the start-up itself, they don't think about the growth of the market. I care much more about the growth rate of the market than its current size, and I also care if there's any reason it's going to top out. You should think about this. I prefer to invest in a company that's going after a small, but rapidly growing market, than a big, but slow-growing market.

One of the big advantages of these sorts of markets - these smaller, rapidly growing markets - is that customers are usually pretty desperate for a solution, and they'll put up with an imperfect, but rapidly improving product. A big advantage of being a student - one of the two biggest advantages - is that you probably have better intuition about which markets are likely to start growing rapidly than older people do. Another thing that students usually don't understand, or it takes awhile, [is that] you can not create a market that does not exist. You can basically change everything in a start-up but the market, so you should actually do some thinking to be sure - or be as sure as you can be - that the market you're going after is going to grow and be there.

There are a lot of different ways to talk about the right kind of market. For example, surfing some one else's wave, stepping into an up elevator, or being part of a movement, but all of this is just a way of saying that you want a market that's going to grow really quickly. It may seem small today, it may be small today, but you know - and other people don't - that it's going to grow really fast.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
So think about where this is happening in the world. You need this sort of tailwind to make a startup successful.

The exciting thing is there are probably more of these tailwinds now than ever before. As Marc Andreessen says, software is eating the world. It's just everywhere, there are so many great ideas out there. You just have to pick one, and find one that you really care about.

Another version of this, that gets down to the same idea, is Sequoia's famous question: Why now? Why is this the perfect time for this particular idea, and to start this particular company. Why couldn't it be done two years ago, and why will two years in the future be too late? For the most successful startups we've been involved with, they've all had a great idea and a great answer to this question. And if you don't you should be at least somewhat suspicious about it.

In general, it's best if you're building something that you yourself need. You'll understand it much better than if you have to understand it by talking to a customer to build the very first version. If you don't need it yourself, and you're building something someone else needs, realize that you're at a big disadvantage, and get very very close to your customers. Try to work in their office, if you can, and if not, talk to them multiple times a day.

Another somewhat counterintuitive thing about good startup ideas is that they're almost always very easy to explain and very easy to understand. If it take more than a sentence to explain what you're doing, that's almost always a sign that it's too complicated. It should be a clearly articulated vision with a small number of words. And the best ideas are usually very different from existing companies, [either] in one important way, like Google being a search engine that worked just really well, and none of the other stuff of the portals, or totally new, like SpaceX. Any company that's a clone of something else, that already exists, with some small or made up differentiator—like X, beautiful design, or Y for people that like red wine instead—that usually fails.

So as I mentioned, one of the great things about being a student is that you've got a very good perspective on new technology. And learning to have good ideas takes a while, so start working on that right now. That's one thing we hear from people all the time, that they wish they had done more of as a student.

The other is meeting potential cofounders. You have no idea how good of an environment you're in right now, for meeting people you can start a company with down the road. And the one thing that we always tell college students is that more important than any particular startup is getting to know potential cofounders.

So I want to finish this section of my talk with a quote from 50 Cent. This is from when he was asked about Vitamin Water. I won't read it, it's up there, but it's about the importance of thinking about what customers want, and thinking about the demands of the market. Most people don't do this—most students especially don't do this. If you can just do this one thing, if you can just learn to think about the market first, you'll have a big leg up on most people starting startups. And this is probably the thing we see wrong with Y Combinator apps most frequently, is that people have not thought about the market first, and what people want first.

So for the next section, I'm going to talk about building a great product. And here, again, I'm going to use a very broad definition of product. It includes customer support, the copy you write explaining the product, anything involved in your customer's interaction in what you built for them.

To build a really great company, you first have to turn a great idea into a great product. This is really
hard, but its crucially important, and fortunately its pretty fun. Although great products are always new to the world, and its hard to give you advice about what to build, there are enough commonalities that we can give you a lot of advice about how to build it.

One of the most important tasks for a founder is to make sure that the company builds a great product. Until you build a great product, nothing else matters. When really successful startup founders tell the story of their early days its almost always sitting in front of the computer working on their product, or talking to their customers. That's pretty much all the time. They do very little else, and you should be very skeptical if your time allocation is much different. Most other problems that founders are trying to solve, raising money, getting more press, hiring, business development, et cetera, these are significantly easier when you have a great product. Its really important to take care of that first. Step one is to build something that users love. At YC, we tell founders to work on their product, talk to users, exercise, eat and sleep, and very little else. All the other stuff I just mentioned—PR, conferences, recruiting advisers, doing partnerships—you should ignore all of that, and just build a product and get it as good as possible by talking to your users.

Your job is to build something that users love. Very few companies that go on to be super successful get there without first doing this. A lot of good-on-paper startups fail because they merely make something that people like. Making something that people want, but only a medium amount, is a great way to fail, and not understand why you're failing. So these are the two jobs

Something that we say at YC a lot is that its better to build something that a small number of users love, then a large number of users like. Of course, it would be best to build something that a small number of users love, but opportunities to do that for v1 are rare, and they're usually not available to startups. So in practice you end up choosing the gray of the orange. You make something that a lot of users like a little bit, or something that a small number of users love a lot. This is a very important piece of advice. Build something that a small number of users love. It is much easier to expand from something that small number of people love, to something that a lot of people love, then from something that a lot of people like to a lot of people love. If you get right, you can get a lot of other things wrong. If you don't get this right, you can get everything else right, and you'll probably still fail. So when you start on the startup, this is the only thing you need to care about until its working.

[Audience member]: Can you go over that slide again?

So you have a choice in a startup. The best thing of all worlds is to build a product that a lot of people really love. In practice, you can't usually do that, because if there's an opportunity like that, Google or Facebook will do it. So there's like a limit to the area under the curve, of what you can build. So you can build something that a large number of users like a little bit, or a small number of users love a lot. So like the total amount of love is the same, its just a question of how its distributed. [audience laughter] And there's like this law of conversation of how much happiness you can put in the world, with the first product of a startup.

And so startups always struggle, with which of those two they should go. And they seem equal, right? Because the area under the curve is the same. But we've seen this time and again, that they're not. And that it's so much easier to expand, once you've got something that some people love, you can expand that into something that a lot of other people love. But if you start with ambivalence, or weak enthusiasm, and try to expand that, you'll never get up to a lot of people loving it. So the advice is: find a small group of users, and make them love what you're doing

One way that you know when this is working, is that you'll get growth by word of mouth. If you get something people love, people will tell their friends about it. This works for consumer product and

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
enterprise products as well. When people really love something, they'll tell their friends about it, and you'll see organic growth.

If you find yourself talking about how it's okay that you're not growing—because there's a big partnership that's going to come save you or something like that—it's almost always a sign of real trouble. Sales and marketing are really important, and we're going to have two classes on them later. A great product is the secret to long term growth hacking. You should get that right before anything else. It doesn't get easier to put off making a great product. If you try to build a growth machine before you have a product that some people really love, you're almost certainly going to waste your time. Breakout companies almost always have a product that's so good, it grows by word of mouth. Over the long run, great product win. Don't worry about your competitors raising a lot of money, or what they might do in the future. They probably aren't very good anyway. Very few startups die from competition. Most die because they themselves fail to make something users love, they spend their time on other things. So worry about this about all else.

Another piece of advice to make something that users love: start with something simple. It's much much easier to make a great product if you have something simple. Even if your eventual plans are super complex, and hopefully they are, you can almost always start with a smaller subset of the problem then you think is the smallest, and its hard to build a great product, so you want to start with as little surface area as possible. Think about the really successful companies, and what they started with, think about products you really love. They're generally incredibly simple to use, and especially to get started using. The first version of Facebook was almost comically simple. The first version of Google was just a webpage with a textbox and two buttons; but it returned the best results, and that's why users loved it. The iPhone is far simpler to use then any smartphone that ever came before it, and it was the first one users really loved.

Another reason that simple's good is because it forces you to do one thing extremely well and you have to do that to make something that people love.

The word fanatical comes up again and again when you listen to successful founders talk about how they think about their product. Founders talk about being fanatical in how they care about the quality of the small details. Fanatical in getting the copy that they use to explain the product just right. and fanatical in the way that they think about customer support. In fact, one thing that correlates with success among the YC companies is the founders that hook up pager duty to their ticketing system, so that even if the user emails in the middle of the night when the founder's asleep, they still get a response within an hour. Companies actually do this in the early days. Their founders feel physical pain when the product sucks and they want to wake up and fix it. They don't ship crap, and if they do, they fix it very very quickly. And it definitely takes some level of fanaticism to build great products.

You need some users to help with the feedback cycle, but the way you should get those users is manually—you should go recruit them by hand. Don't do things like buy Google ads in the early days, to get initial users. You don't need very many, you just need ones that will give you feedback everyday, and eventually love your product. So instead of trying to get them on Google Adwords, just the few people, in the world, that would be good users. Recruit them by hand.

Ben Silbermann, when everyone thought Pinterest was a joke, recruited the initial Pinterest users by chatting up strangers in coffee shops. He really did, he just walked around Palo Alto and said "Will you please use my product?" He also used to run around the Apple store in Palo Alto, and he would like set all the browsers to the Pinterest homepage real quick, before they caught him and kicked him out. (laughter) and so that when people walked in they were like "Oh, what's this?". This is an important example of doing things that don't scale. If you haven't read Paul Graham's essay on that topic, you
definitely should.

So get users manually and remember that the goal is to get a small group of them to love you. Understand that group extremely well, get extremely close to them. Listen to them and you'll almost always find out that they're very willing to give you feedback. Even if you're building the product for yourself, listen to outside users, and they'll tell you how to make a product they'll pay for. Do whatever you need to make them love you, and make them know what you're doing. Because they'll also be the advocates that help you get your next users.

You want to build an engine in the company that transforms feedback from users into product decisions. Then get it back in from of the users and repeat. Ask them what they like and don't like, and watch them use it. Ask them what they'd pay for. Ask them if they'd be really bummed if your company went away. Ask them what would make them recommend the product to their friends, and ask them if they'd recommended it to any yet.

You should make this feedback loop as tight as possible. If your product gets 10 times better every week, that compounds really quickly. One of the advantages of software startups is just how short you can make the feedback loop. It can be measured in hours, and the best companies usually have the tightest feedback loop. You should try to keep this going for all of your company's life, but it's really important in the early days.

The good news is that all this is doable. It's hard, it takes a lot of effort, but there's no magic. The plan is at least is straightforward, and you will eventually get to a great product.

Great founders don't put anyone between themselves and their users. The founders of these companies do things like sales and customer support themselves in the early days. It's critical to get this loop embedded in the culture. In fact, a specific problem we always see with Stanford startups, for some reason, is that the students try to hire sales and customer support people right away, and you've got to do this yourself, it's the only way.

You really need to use metrics to keep yourself honest on this. It really is true that the company will build whatever the CEO decides to measure. If you're building an Internet service, ignore things like total registrations—don't talk about them, don't let anyone in the company talk about them—and look at growth and active users, activity levels, cohort retention, revenue, net promoter scores, these things that matter. And then be brutally honest if they're not going in the right direction. Startups live on growth, it's the indicator of a great product.

So this about wraps up the overview on building a great product. I want to emphasize again, that if you don't get this right, nothing else we talk about in the class will matter. You can basically ignore everything else in the class until this is working well. On the positive side, this is one of the most fun parts of building a startup.

So I'm going to pause here, we'll pick back up with the rest of this on Thursday, and now Dustin is going to talk about why you should start a startup. Thank you for coming, Dustin.
Lecture 1: Why To Start A Startup

http://startupclass.samaltman.com/courses/lec01/

Dustin Moskovitz

But yeah, Sam asked me to talk about why you should start a startup. There's a bunch of common reasons that people have, that I hear all the time for why you might start a startup. Its important to know what reason is yours, because some of them only make sense in certain contexts, some of them will actually, like, lead you astray. You may have been mislead by the way that Hollywood or the press likes to romanticize entrepreneurship, so I want to try to illuminate some of those potential fallacies, so you guys can make the decision in a clear way. And then I'll talk about the reason I like best for actually starting a startup, its very related to a lot of what Sam just talked about. But surprisingly, I don't think its the most common reason. Usually people have one of these other reasons, or, you know, they just want to start a company for the sake of starting a company.

So the 4 common reasons, just to enumerate them, are it's glamorous, you'll get to be the boss, you'll have flexibility, especially over your schedule, and you'll have the chance to have bigger impact and make more money then you might by joining a later stage company.

So you guys are probably pretty familiar this concept, when I wrote the Medium post, which a lot of you guys read a year ago, I felt like the story in the press was a little more unbalanced, entrepreneurship got romanticized quite a bit. The movie The Social Network came out, it had a lot of like bad aspects of what it like to be an entrepreneur, but mainly it painted this picture of like, there's a lot of partying and you just kind of move from like one brilliant insight to another brilliant insight, and really made it seem like this really cool thing to do.

And I think the reality is just not quite so glamorous, there's an ugly side to being an entrepreneur, and more importantly, what you're actually spending your time on is just a lot of hard work. Sam mentioned this, but your basically just sitting at your desk, heads down, focused, answering customer support emails, doing sales, figuring out hard engineering problems. So its really important that you go in with eyes wide open. And then its also quite stressful. This has been a popular topic in the press lately: The Economist actually ran a story just last week called "Entrepreneurs anonymous", and shows a founder like hiding under his desk, talking about founder depression. So this is a very real thing. Let's be real, if you start a company its going to be extremely hard.

Why is it so stressful? So a couple reasons. One is you've got a lot of responsibility. People in any career have a fear of failure, its kind of just like a dominant part of the part of the psychology. But when you're an entrepreneur, you have fear of failure on behalf of yourself and all of the people who decided to follow you. So that's really stressful. In some cases people are depending on you for their livelihood, even when that's not true, they've decided to devote the best years of their life to following you. So you're responsible for the opportunity cost of their time. You're always on call, if something comes up—maybe not always at 3 in the morning, but for some startups that's true—but if something important comes up, you're going to deal with it. That's kinda the end of the story, doesn't matter if you're on vacation, doesn't matter if its the weekend, you've got to always be on the ball and be in a place mentally where you're prepared to deal with those things. A sort of special example of this kind of stress is fundraising.

So a scene from The Social Network. This is us partying and working at the same time—somebody's spraying champagne everywhere—The Social Network spends a lot of time painting these scenes. Mark's
not in the scene, the other thing they spend all their time on is painting him out to be a huge jerk.

This is an actual scene from Palo Alto, he spent a lot of time at this desk, head down and focused. Mark was still kinda a jerk sometimes, but in this more like fun lovable way, and not in a sociopathic, scorned lover way. So this is just him signaling his intention to just be focused and keep working, not be social.

So then there's the scene demonstrating the insight moment, it's kind of like out of A Beautiful Mind, they literally stole that scene. So they like to paint that scene and jump to these moments from other moments, with partying in between. But really we were just at that table the whole time. So if you compare this photo, Mark is in the exact same position but he's wearing different clothes, so this is definitely a different day. That's what it's actually like in person. I just covered this bullet; this is the Economist article I was talking about a second ago.

So another form of stress is unwanted media attention. So part of it being glamorous is you get some positive media attention sometimes, it's nice to be on the cover of Time and to be the Person of the Year. It's maybe a little less nice to be on the cover of People with one of your wedding photos. It depends on who you are, I really hate it, but when Valleywag analyzes your lecture and tears you apart, you don't want that, you definitely don't want that. Nobody wants that.

One thing I almost never hear people talk about is you're much more committed. So if you're at a startup and it's very stressful and things are not going well, you're unhappy, you can just leave. For a founder, you can leave, but it's very uncool and pretty much a black eye for the rest of your career. And so you really are committed for ten years if it's going well and probably more like five years if it's not going well. So three years to figure out it's not going well and then if you find a nice landing for your company, another two years at the acquiring company. If you leave before that, again it's not only going to harm yourself financially but it's going to harm all your employees. So if you're lucky and you have a bad startup idea, you fail quickly, but most of the time it's not like that.

I should say, I've had a lot of this stress in my own life, especially in the early years of Facebook, I got really unhealthy, I wasn't exercising, I had a lot of anxiety actually throughout my back, like almost every six months, when I was twenty-one or twenty-two, which is pretty crazy. So if you do start a company, be aware that you're going to deal with this. You're going to have to actually manage this, it's one of your core responsibilities. Ben Horowitz likes to say the number one role of a CEO is managing your own psychology, it's absolutely true, make sure you do it.

Another reason, especially if you've had another job at another company, you start to develop this narrative, like the people running this company are idiots, they're making all these decisions and spending all their time in these stupid ways, I'm gonna start a company and I'm going to do it better. I'm going to set all the rules.

Sounds good, makes a lot of sense. If you've read my media post, you'll know what's coming, I'll give you guys a second to read this quote:

*People have this vision of being the CEO of a company they started and being on top of the pyramid. Some people are motivated by that, but that's not at all what it's like.*

*What it's really like: everyone else is your boss – all of your employees, customers, partners, users, media are your boss. I've never had more bosses and needed to account for more people today.*
The life of most CEOs is reporting to everyone else, at least that's what it feels like to me and most CEOs I know. If you want to exercise power and authority over people, join the military or go into politics. Don't be an entrepreneur.

-Phil Libin

This really resonates with me. One thing to point out is that the reality of these decision is nuanced. The people you thought were idiots probably weren't idiots, they just had a really difficult decision in front of them and people pulling them in multiple directions. So the most common thing I have to spend my time on and my energy on as a CEO is dealing with the problems that other people are bringing to me, the other priorities that people create, and it's usually in the form of a conflict. People want to go in different directions or customers want different things. And I might have my own opinions on that, but the game I'm playing is who do I disappoint the least and just trying to navigate all these difficult situations.

And even on a day to day basis, I might come in on Monday and have all these grand plans for how I'm going to improve the company. But if an important employee is threatening to quit, that's my number one priority. That's what I'm spending my time on.

A subset of You're the Boss is you have flexibility, you have control over your own schedule. This is a really attractive idea. So here's the reality:

If you're going to be an entrepreneur, you will actually get some flex time to be honest. You'll be able to work any 24 hours a day you want!

- Phil Libin

This truly resonates with me as well. Some of the reasons for this again, you're always on call. So maybe you don't intend to work all parts of the day, but you don't control which ones.

You're a role model of the company, and this is super important. So you might have some good weeks and you might have some bad weeks, some weeks when you're low energy and you might want to take a couple days off. That's really bad if you're an entrepreneur. Your team will really signal off of what you're bringing to the table. So if you take your foot off the gas, so will they.

You're always working anyways. If you're really passionate about an idea, it's going to pull you towards it. If you're working with great investors, you're working with great partners, they're going to be working really hard, they're going to want you to be working really hard.

Some companies like to tell the story about you can have your cake and eat it too, you can have like 4 days work weeks maybe, if you're Tim Ferris maybe you can have a 12 hours work week. It's a really attractive idea and it does work in a particular instance which is if you wanna actually have a small business to go after in each market then you are a small business entrepreneur, that makes little sense but as soon as you get past like 2 or 3 people you really need to step it up and be full-time committed.

You'll make more money and have more impact

This is the big one, the one I hear the most especially like candidates applying to a [?], they tell me "You know I'd really like to work for much smaller companies or start my own because then I have a
much bigger slice of the pie or have much more impact on how that company does and I'll have more equity so I'll make more money as well". So let's examine when this might be true.

**Financial Reward & Impact**

<table>
<thead>
<tr>
<th></th>
<th>A Real Company</th>
<th>Employee #100’s Upside (10bp)</th>
<th>Your First Company</th>
<th>Theoretical Valuation</th>
<th>Founder’s Upside (10%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dropbox</strong></td>
<td>$10B</td>
<td>$10M</td>
<td>“Uber for Pet Sitting”</td>
<td>$100M</td>
<td>$10M</td>
</tr>
<tr>
<td><strong>Facebook</strong></td>
<td>$200B</td>
<td>$200M</td>
<td>“Uber for Space Travel”</td>
<td>$2B</td>
<td>$200M</td>
</tr>
</tbody>
</table>

**CS 183B: Dustin Moskovitz**

I'll explain these tables. They're a little complex but let's focus on the left first. These are just explaining Dropbox and Facebook, these are their current valuations and this is how much money you might make as employee number 100 coming into these companies especially if you're like an experienced, relatively experienced engineer, you have like 5 years of industry experience, you're pretty likely to have an offer that's around 10 base points. If you joined Dropbox couple years ago the upside you've already locked in is about $10M and there's plenty more growth from there. If you joined Facebook a couple years into its existence you've already made around $200M, this is a huge number and even if you joined Facebook as employee number 1000, so you joined like 2009, you still make $20M, that's a giant number and that's how you should be benchmarking when you're thinking about what you might make as an entrepreneur.

Moving over to the table on the right, these are two theoretical companies you might start. "Uber for Pet Sitting", pretty good idea if you're really well suited to this you might have a really good shot at building a $100M company and your share of that company is likely to be around 10%; that certainly fluctuates a lot, some founders have more than this, some founders have a lot less, but after multiple rounds of dilution, multiple rounds of option pool creation you're pretty likely to end up about here. If you have more than this I'd recommend Sam's post on equity split between founders and employees, you should be probably giving out more.

So basically if you're extremely confident in building a $100M, which is a big ask, it should go without
saying that you should have a lot more confidence on Facebook in 2009 or Dropbox in 2014 that you might for a startup that doesn't even exist yet, then this is worth doing. If you have a $100M idea and you're pretty confident you can execute it I'd consider that.

If you think you're the right entrepreneur to build "Uber for Space Travel", that's a really huge idea, $2B idea, you're actually gonna have a pretty good return for that, you should definitely do that, this is also the value only after 4 years and this idea probably has legs, definitely go after that, if you're thinking of building that you probably shouldn't even be in this class right now, just go build that company.

So why is this financial reward and impact? I really think that financial reward is very strongly correlated with the impact we have on the world, if you don't believe that let's talk through some specific examples and not think about the equity at all.

So why might joining a late stage company actually might have a lot of impact, you get this force multiplier: they have an existing mass of user base, if it's Facebook it's a billion users, if it's Google it's a billion users, they have existing infrastructures you get to build on, that's also increasingly true for a new startup like AWS and all these awesome independent service providers, but you usually get some micro-proprietary technology and they maintain it for you, it's a pretty great place to start. And you get to work with a team, it'll help you leverage your ideas into something great.

So couple specific examples, Bret Taylor came into Google as around employee number 1500 and he invented Google Maps, that's a product you guys probably use everyday, I used it to get here and it's used by hundreds of millions of people around the world. He didn't need to start a company to do that, he happened to get a big financial reward, but the point is yet again massive impact.

My cofounder Justin Rosenstein joined Google a little later after Brett, he was a PM there and just as a side project he ended up prototyping a chat which used to be a stand-alone app, integrated in Gmail like you see in the upper right there and before he did that like you couldn't even think you could chat over Ajax or chat in the browser at all and he just kinda demonstrated it and showed it to his team and made it happen. This is probably a product most of you use almost everyday.

Perhaps even more impressively, shortly after that Justin left and became employee around 250 at Facebook and he led a hackaton project along with people like Andrew Bosworth and [?] to create the Like button, this is one of the most popular elements anywhere on the web, totally changed how people use it and then again didn't need to start a company to do it and almost certainly would have failed if he had tried because he really needed the distribution of Facebook to make it work.

So important to keep in mind the context for what kind of company you're trying to start and like where you will actually be able to make it happen.

So what's the best reason?

Sam already talked about this a little bit, but basically you can't not do it. You're super passionate about this idea, you're the right person to do it, you've gotta make it happen. So how does this break down?

This is a wordplay, you can't not do it in two ways. One is you're so passionate about it that you have to do it and you're going to do it anyways. This is really important because you'll need that passion to get through all of those hard parts of being an entrepreneur that we talked about earlier. You'll also need it to effectively recruit, candidates can smell when you don't have passion and there are enough
entrepreneurs out there that do have passion so they may as well work for one of those! So this is table stakes for being an entrepreneur. Your subconscious can also tell when you don't have passion and that can be a huge problem.

The other way to interpret this is the world needs you to do it. This is validation that the idea is important, that it's going to make the world better, so the world needs it. If it's not something the world needs, go do something the world needs. Your time is really valuable, there are plenty of good ideas out there, maybe it's not your own, maybe it's at an existing company, but you may as well work on something that's going to be good.

The second way to interpret this is that the world needs you to do it. You're actually well suited for this problem in some way. If this isn't true, it may be a sign that your time is better spent somewhere else. But best case scenario if this isn't true, you outcompete the team for which it is true and it's a suboptimal outcome for the world and that doesn't feel very good.

So drawing this back to my own experience at Asana, Justin and I were reluctant entrepreneurs before we founded Asana, we were working at Facebook and we were working on a great problem. We would basically work all day long on our normal projects and then at night we would keep working on this internal task manager that was used internally at the company and it was just because we were so passionate about the idea, it was so clearly valuable that we couldn't do anything else.

And at some point we had to have the hard conversation of okay what does it mean if we don't actually start this company. We could see the impact it was having at Facebook, we were convinced it was valuable to the world. We were also convinced no one else was going to build it, the problem had been around a long time and we just kept seeing incremental solutions to it and so we believed if we didn't come out with the solution we thought was best, there would be a lot of value left on the table. We couldn't stop working on it and literally the idea was beating itself out of our chests and forcing itself out into the world. And I think that's really the feeling you should be looking for when you start a company, that's how you know you have the right idea.

I'll go ahead and stop there. I'll put some recommended books up here.

**Recommended Reading**

- The Hard Thing About Hard Things
- Zero to One (CS 138A)
- The Facebook Effect
- The 15 Commitments of Conscious Leadership
- The Tao of Leadership
- Nonviolent Communication

Thank you.
Lecture 1: Advice for Ambitious 19 Year Olds


“I’m an ambitious 19 year old, what should I do?”

I get asked this question fairly often, and I now have a lot of data on what works, so I thought I’d share my response.

Usually, people are deciding between going to college (and usually working on side projects while they do so), joining a company, or starting their own startup. [1] [2]

The secret is that any of these can be right answer, and you should make your decision based on the specific circumstances of each option. The critical point is that you want to do the thing that is most likely to get you on a path to do something great.

No matter what you choose, build stuff and be around smart people. “Stuff” can be a lot of different things—open source projects outside of class, a startup, a new sales process at a company you work at—but, obviously, sitting around talking with your friends about how you guys really should build a website together does not count.

The best people always seem to be building stuff and hanging around smart people, so if you have to decide between several options, this may be a good filter.

Working on something good will pull you along a path where good things keep happening to you. (In fact, this effect is so strong that there’s a danger of getting sucked into too many interesting things and getting distracted from what you really want to do.)

In making this decision, you want to take the right kind of risk. Most people think about risk the wrong way—for example, staying in college seems like a non-risky path. However, getting nothing done for four of your most productive years is actually pretty risky. Starting a company that you’re in love with is the right kind of risk. Becoming employee number 50 at a company that still has a good chance of failure is the wrong kind of risk.

If you stay in college, make sure you learn something worthwhile and work on interesting projects—college is probably the best place to meet people to work with. If you’re really worried you’ll miss some critical social experience by dropping out of college, you should probably stay.

If you join a company, my general advice is to join a company on a breakout trajectory. There are a usually a handful of these at a time, and they are usually identifiable to a smart young person. They are a very good risk/reward tradeoff. Such a company is almost certainly going to be successful, but the rest of the world isn’t quite as convinced of it as they should be. Fortunately, these companies love ambitious young people. In addition to the equity being a great deal (you might get 1/10th of the equity you’d get if you going a tiny new startup, but at 1/100th or 1/1000th of the risk), you will work with very good people, learn what success looks like, and get a W on your record (which turns out to be quite valuable). Spending a few years at a company that fails has path consequences, and working at an already-massively-successful company means you will learn much less, and probably work with less impressive people.

Incidentally, don’t let salary be a factor. I just watched someone turn down one of these breakout companies because Microsoft offered him $30k per year more in salary—that was a terrible decision. He will not build interesting things and may not work with smart people. In a few years, when it’s time for something new, the options in front of him will be much worse than they could have been.

If you start a company, only do so if you have an idea you’re in love with. If you’re hanging out with your friends trying to come up with an idea, I don’t think you should start that company (although there are many who disagree with me). Starting a failed startup is less bad than joining a failed company as an employee (and you’ll certainly learn much more in the former case). If you fail at an idea that you really loved and could have been great, you’re unlikely to regret it, and people will not hold it against you. Failing at a me-too copycat startup is worse. Remember that there will be lots of other opportunities to start companies, and that startups are a 6-10 year commitment—wait for the right one.

One big pro for starting a company is that it’s usually the way to learn the most in the shortest amount of time. One big con is that it’s easy to start a company for the wrong reasons—usually so that you can say you’re starting a company—and this makes it easy to cloud your judgment.

No matter what you choose, keep your personal burn rate low and minimize your commitments. I have seen a lot of people miss great opportunities because they couldn’t afford a reduction in salary or because they couldn’t move or didn’t have the time.

Think about risk the right way. Drew Houston gave a great commencement speech where he said you only have to be right once. That’s true. The risk is not getting on the path where you get to be right that one critical time.

Thanks to Lachy Groom and Nick Sivo for reading drafts of this.

[1] Sometimes, a 4th option is being a VC. This is usually a mistake—the best way to become a VC is not to grind your way up the ladder from junior associate intern. Even if you want to be a VC, you’re much better off starting or joining a startup, and getting partner offers when you’re 28. Plus,
good founders want to work with an investor that has operational experience.

[2] Interestingly, no one is ever considering going into academia.

Lecture 1: Good and Bad Reasons to Become an Entrepreneur


Dustin Moskovitz

Recently we hosted a Q&A at Asana that I participated in with Ben Horowitz, Matt Cohler, and Justin Rosenstein. Marcus Wohlsen from Wired attended and wrote an article that discussed our views on the culture of entrepreneurship in Silicon Valley. This is an important topic, so I want to take some time to clarify what we meant in this blog post. Before I do, I’d like to emphasize that we were talking exclusively about Silicon Valley culture and not the more general ‘small business entrepreneur.’ So for our audience at the time, entrepreneur meant “Silicon Valley startup technology entrepreneur.”

Even given that context, it is notable that we all said you “probably” shouldn’t be an entrepreneur, not that you definitely shouldn’t. This is explicitly a directional position; we believe there are too many startups and entrepreneurs in the SV ecosystem, but that is very different from saying there shouldn’t be any. Many people think there should be more, and we are counterbalancing that view. Whenever you counterbalance an extreme view, you tend to also come off extremely, and certainly do in the media (which is related to the point I made about integration in my last post).

The reason we like best for becoming an entrepreneur is that you are extremely passionate about an idea and believe that starting a new company is the best way to bring it into the world. The passion is important because entrepreneurship is hard and you’ll need it to endure the struggle, as well as to convince other people to help you. Believing that starting a new company is the best way to bring it into the world is important to ensure that resources—including most importantly your own time—are being put to the best possible use. If the idea is best brought into the world by an existing team, then it is tautologically optimal for the world for it to happen that way. Of course, not everyone is actually trying to optimize their impact, but many entrepreneurs are, by their own admission, and it is important for those people to consider this angle.

If you’re not trying to maximize impact, then it seems like a reasonable assumption that you are instead optimizing around personal lifestyle preferences of some kind. You want total freedom to choose how you make your living, regardless of if it necessarily provide large amounts of value to other people or perhaps is even redundant with something that already exists. Or you want extreme flexibility in your schedule, maybe including the ability to stop working altogether for long periods of time at short notice. Or you want to work on a certain kind of problem or with certain kinds of people. For many kinds of preference, you likely can actually find a company able to give them to you, but
Certainly starting your own is a great shortcut and I personally think that's totally reasonable. I like people who are seeking to have big impact on the world, but it is not the only path worth taking, and I have no reason to denigrate this type of entrepreneur.

So with all that in mind, here are some of the bad reasons to become an entrepreneur that we were actually trying to speak to:

You want to be your own boss in a big company. Evernote CEO Phil Libin put it well last year:

“People have this vision of being the CEO of a company they started and being on top of the pyramid. Some people are motivated by that, but that’s not at all what it’s like.

* What it’s really like: everyone else is your boss – all of your employees, customers, partners, users, media are your boss. I’ve never had more bosses and needed to account for more people today.*

*The life of most CEOs is reporting to everyone else, at least that’s what it feels like to me and most CEOs I know. If you want to exercise power and authority over people, join the military or go into politics. Don’t be an entrepreneur.*”

- You think it’s glamorous. The media does a great job idolizing various entrepreneurs, crowning Kings and designating Godfathers of various mafias, but this is all colorful narrative. The reality is years of hard work, throughout which you usually have no idea if you’re even moving in the right direction.

- You believe you’re extremely talented and that this is the way to maximize your financial return on that talent. Why wouldn’t you want more of the cap table? This is flawed logic, since the 100th engineer at Facebook made far more money than 99% of Silicon Valley entrepreneurs. Small slices of gigantic pies are still themselves gigantic. If you’re extremely talented, you can easily identify a company with high growth potential and relatively low risk and get an aggressive compensation package from them. If you turn out to be wrong after a few years, you can try again. Within 2 or 3 tries, and likely on the first one, you’ll have a great outcome and can be confident you contributed serious lasting value to the world. If you instead try to immediately start “the next Google or Facebook”, there is a very high likelihood that you will fail completely, or be forced to settle for a much smaller outcome. It will take a long time to reach success or failure, so you won’t have many tries.

- You heard Paul Graham compared non-founders to animals at a zoo and don’t want to look like a chump (or a chimp). As Jeff Atwood points out, this is much more projection than fact. There are companies that feel like they were lifted out of a Dilbert cartoon and treat their employees terribly and then there are, you know, good companies. Work for one of those, or work for yourself, but definitely don’t work anywhere that you can describe as “soul-sucking”.

Yes, this viewpoint is somewhat self-serving. We want to hire the best at Asana, and many of the best are choosing to become entrepreneurs instead of applying at companies like ours. That said, we only need to hire a quite small number of those folks to ourselves succeed. The bigger concern is the macro effect of spreading talent too thinly across the board and the micro effect of people not spending their time in valuable ways. Zuck himself has speculated that Silicon Valley is not obviously the place he would start a company like Facebook today for exactly this reason. As to the micro effect, I like the way Justin puts it: “If you’re going to devote the best years of your life to your work, have enough love for yourself and the world around you to work on something that matters to you deeply.”

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
2. Ideas, products, teams and execution II
Before I jump into today's lecture, I wanted to answer a few questions people had emailed me about the last lecture that we didn't have time for. So, if you have a question about what we covered last time, I am welcome to answer it now, starting with you.

Q: How do I identify if a market has a fast growth rate now and also for the next ten years?

A: The good news about this is this is one of the big advantages students have. You should just trust your instincts on this. Older people have to basically guess about the technologies young people are using. But you can just watch what you're doing and what your friends are doing and you will almost certainly have better instincts than anybody older than you. And so the answer to this is just trust your instincts, think about what you're doing more, think about what you're using, what you're seeing people your age using, that will almost certainly be the future.

Okay, one more question on the last lecture before we start.

Q: How do you deal with burnout while still being productive and remaining productive.

A: The answer to this is just that it sucks and you keep going. Unlike a student where you can throw up your hands and say you know I'm really burnt out and I'm just going to get bad grades this quarter, one of the hard parts about running a startup is that it's real life and you just have to get through it. The canonical advice is to go on a vacation and that never works for founder. It's sort of all consuming in this way that is very difficult to understand.

So what you do is you just keep going. You rely on people, it's really important, founder depression is a serious thing and you need to have a support network. But the way through burn out is just to address the challenges, to address the things that are going wrong and you'll eventually feel better.

Last lecture, we covered the idea and the product and I want to emphasize that if you don't get those right, none of the rest of this is going to save you. Today, we're going to talk about how to hire and how to execute. Hopefully you don't execute the people you hire. Sometimes.

First, I want to talk about cofounders. Cofounder relationships are among the most important in the entire company. Everyone says you have to watch out for tension brewing among cofounders and you have to address is immediately. That's all true and certainly in YC's case, the number one cause of early death for startups is cofounder blowups. But for some reason, a lot of people treat choosing their cofounder with even less importance than hiring. Don't do this! This is one of the most important decisions you make in the life of your startup and you need to treat it as such.

And for some reason, students are really bad at this. They just pick someone. They're like, I want to start a business and you want to start a business, let's start a startup together. There are these cofounder dating things where you're like, Hey I'm looking for a cofounder, we don't really know each other, let's start a company. And this is like, crazy. You would never hire someone like this and yet people are willing to choose their business partners this way. It's really really bad. And choosing a random random cofounder, or choosing someone you don't have a long history with, choosing someone you're not
friends with, so when things are really going wrong, you have this sort of past history to bind you
together, usually ends up in disaster.

We had one YC batch in which nine out of about seventy-five companies added on a new cofounder
between when we interviewed the companies and when they started, and all nine of those teams fell
apart within the next year. The track record for companies where the cofounders don't know each other
is really bad.

A good way to meet a cofounder is to meet in college. If you're not in college and you don't know a
cofounder, the next best thing I think is to go work at an interesting company. If you work at Facebook
or Google or something like that, it's almost as cofounder rich as Stanford. It's better to have no
cofounder than to have a bad cofounder, but it's still bad to be a solo founder. I was just looking at the
stats here before we started. For the top, and I may have missed one because I was counting quickly,
but I think, for the top twenty most valuable YC companies, almost all of them have at least two
founders. And we probably funded a rate of like one out of ten solo teams.

So, best of all, cofounder you know, not as good as that, but still okay, solo founder. Random founder
you meet, and yet students do this for some reason, really really bad.

So as you're thinking about cofounders and people that could be good, there's a question of what you're
looking for right? At YC we have this public phrase, and it's relentlessly resourceful, and everyone's
heard of it. And you definitely need relentlessly resourceful cofounders, but there's a more colorful
example that we share at the YC kickoff. Paul Graham started using this and I've kept it going.

So, you're looking for cofounders that need to be unflappable, tough, they know what to do in every
situation. They act quickly, they're decisive, they're creative, they're ready for anything, and it turns out
that there's a model for this in pop culture. And it sounds very dumb, but it's at least very memorable
and we've told every class of YC this for a long time and I think it helps them.

And that model is James Bond. And again, this sounds crazy, but it will at least stick in your memory
and you need someone that behaves like James Bond more than you need someone that is an expert in
some particular domain.

As I mentioned earlier, you really want to know your cofounders for awhile, ideally years. This is
especially true for early hires as well, but incidentally, more people get this right for early hires than
they do for cofounders. So, take advantage of school. In addition to relentlessly resourceful, you want a
tough and a calm cofounder. There are obvious things like smart, but everyone knows you want a smart
cofounder, they don't prioritize things like tough and calm enough, especially if you feel like you
yourself aren't, you need a cofounder who is. If you aren't technical, and even if most of the people in
this room feel like they are, you want a technical cofounder. There's this weird thing going on in
startups right now where it's become popular to say, You know what, we don't need a technical
cofounders, we're gonna hire people, we're just gonna be great managers.

That doesn't work too well in our experience. Software people should really be starting software
companies. Media people should be starting media companies. In the YC experience, two or three
cofounders seems to be about perfect. One, obviously not great, five, really bad. Four works sometimes,
but two or three I think is the target.

The second part of how to hire: try not to. One of the weird things you'll notice as you start a company,
is that everyone will ask you how many employees you have. And this is the metric people use to judge
how real your startup is and how cool you are. And if you say you have a high number of employees,
they're really impressed. And if you say you have a low number of employees, then you sound like this little joke. But actually it sucks to have a lot of employees, and you should be proud of how few employees you have. Lots of employees ends up with things like a high burn rate, meaning you're losing a lot of money every month, complexity, slow decision making, the list goes on and it's nothing good.

So you want to be proud of how much you can get done with a small numbers of employees. Many of the best YC companies have had a phenomenally small number of employees for their first year, sometimes none besides the founders. They really try to stay small as long as they possibly can. At the beginning, you should only hire when you desperately need to. Later, you should learn to hire fast and scale up the company, but in the early days the goal should be not to hire. And one of the reasons this is so bad, is that the cost of getting an early hire wrong is really high. In fact, a lot of the companies that I've been very involved with, that have had a very bad early hire in the first three or so employees never recover, it just kills the company.

Airbnb spent five months interviewing their first employee. And in their first year, they only hired two. Before they hired a single person, they wrote down a list of the culture values that they wanted any Airbnb employee to have. One of those what that you had to bleed Airbnb, and if you didn't agree to that they just wouldn't hire you. As an example of how intense Brian Chesky is, he's the Airbnb CEO, he used to ask people if they would take the job if they got a medical diagnosis that they have one year left to life. Later he decided that that was a little bit too crazy and I think he relaxed it to ten years, but last I heard, he still asks that question.

These hires really matter, these people are what go on to define your company, and so you need people that believe in it almost as much as you do. And it sounds like a crazy thing to ask, but he's gotten this culture of extremely dedicated people that come together when the company faces a crisis. And when the company faced a big crisis early on, everyone lived in the office, and they shipped product every day until the crisis was over. One of the remarkable observations about Airbnb is that if you talk to any of the first forty or so employees, they all feel like they were a part of the founding of the company.

But by having an extremely high bar, by hiring slowly ensures that everyone believes in the mission, you can get that. So let's say, you listened to the warning about not hiring unless you absolutely have too. When you're in this hiring mode, it should be your number one priority to get the best people. Just like when you're in product mode that should be your number one priority. And when you're in fundraising mode, fundraising is your number one priority.

On thing that founders always underestimate is how hard it is to recruit. You think you have this great idea and everyone's going to join. But that's not how it works. To get the very best people, they have a lot of great options and so it can easily take a year to recruit someone. It's this long process and so you have to convince them that your mission is the most important of anything that they're looking at. This is another case of why it's really important to get the product right before looking at anything else. The best people know that they should join a rocketship.

By the way, that's my number one piece of advice if you're going to join a startup, is pick a rocketship. Pick a company that's already working and that not everyone yet realizes that, but you know because you're paying attention, that it's going to be huge. And again, you can usually identify these. But good people know this, and so good people will wait, to see that you're on this trajectory before they join.

One question that people asked online this morning was how much time you should be spending on hiring. The answer is zero or twenty-five percent. You're either not hiring at all or it's probably your single biggest block of time. In practice, all these books on management say you should spend fifty percent of your time hiring, but the people that give that advice, it's rare for them to even spend ten
percent themselves. Twenty-five percent is still a huge amount of time, but that's really how much you should be doing once you're in hiring mode.

If you compromise and hire someone mediocre you will always regret it. We like to warn founders of this but no one really feels it until they make the mistake the first time, but it can poison the culture. Mediocre people at huge companies will cause some problems, but it won't kill the company. A single mediocre hire within the first five will often in fact kill a startup.

A friend of mine has a sign up in the conference room that he uses for interviews and he positions the sign that the candidate is looking at it during the interview and it says that mediocre engineers do not build great companies. Yeah that's true, it's really true. You can get away with it in a big company because people just sort of fall through the cracks but every person at a startup sets the tone. So if you compromise in the first five, ten hires it might kill the company. And you can think about that for everyone you hire: will I bet the future of this company on this single hire? And that's a tough bar. At some point in the company, when you're bigger, you will compromise on a hire. There will be some pressing deadline or something like that you will still regret. But this is the difference between theory and practice we're going to have later speakers talk about what to do when this happens. But in the early days you just can't screw it up.

Sources of candidates. This is another thing that students get wrong a lot. The best source for hiring by far is people that you already know and people that other employees in the company already know. Most great companies in text have been built by personal referrals for the first hundred employees and often many more. Most founders feel awkward but calling anyone good that they've ever met and asking their employees to do the same. But she'll notice if you go to work at Facebook or Google one of the things they do in your first few weeks is an HR person sits you down and beat out of you every smart person you've ever met to be able to recruit them.

These personal referrals really are the trick to hiring. Another tip is to look outside the valley. It is brutally competitive to hire engineers here but you probably know people elsewhere in the world that would like to work with you.

Another question that founders ask us a lot about his experience and how much that matters. The short version here is that experience matters for some roles and not for others. When you're hiring someone that is going to run a large part of your organization experience probably matters a lot. For most of the early hires that you make at a startup, experience probably doesn't matter that much and you should go for aptitude and belief in what you're doing. Most of the best hires that I've made in my entire life have never done that thing before. So it's really worth thinking, is this a role where I care about experience or not. And you'll often find to don't, especially in the early days.

There are three things I look for in a hire. Are they smart? Do they get things done? Do I want to spend a lot of time around them? And if I get an answer, if I can say yes to all three of these, I never regret it, it's almost always worked out. You can learn a lot about all three of these things in an interview but the very best way is working together, so ideally someone you've worked together with in the past and in that case you probably don't even need an interview. If you haven't, then I think it's way better to work with someone on a project for a day or two before hiring them. You'll both learn a lot they will too and most first-time founders are very bad interviewers but very good at evaluating someone after they've worked together.

So one of the pieces of advice that we give at YC is try to work on a project together instead of an interview. If you are going to interview, which you probably will, you should ask specifically about projects that someone worked on in the past. You'll learn a lot more than you will with brainteasers. For
some reason, young technical cofounders love to ask brainteasers rather than just ask what someone has done. Really dig in to projects people have worked on. And call references. That is another thing that first time founders like to skip. You want to call some people that these people have worked with in the past. And when you do, you don't just want to ask, How was so-and-so, you really want to dig in. Is this person in the top five percent of people you've ever worked with? What specifically did they do? Would you hire them again? Why aren't you trying to hire them again? You really have to press on these reference calls.

Another thing that I have noticed from talking to YC companies is that good communication skills tend to correlate with hires that work out. I used to not pay attention to this. We're going to talk more about why communication is so important in an early startup. If someone is difficult to talk to, if someone cannot communicate clearly, it's a real problem in terms of their likelihood to work out. Also, for early employees you want someone that has somewhat of a risk-taking attitude. You generally get this, otherwise they wouldn't be interested in a startup, but now that startups are sort of more in fashion, you want people that actually sort of like a little bit of risk. If someone is choosing between joining McKinsey or your startup it's very unlikely they're going to work out at the startup.

You also want people who are maniacally determined and that is slightly different than having a risk tolerant attitude. So you really should be looking for both. By the way, people are welcome to interrupt me with questions as stuff comes up.

There is a famous test from Paul Graham called the animal test. The idea here is that you should be able to describe any employee as an animal at what they do. I don't think that translates out of English very well but you need unstoppable people. You want people that are just going to get it done. Founders who usually end up being very happy with their early hires usually end up describing these people as the very best in the world at what they do.

Mark Zuckerberg once said that he tries to hire people that A. he'd be comfortable hanging with socially and B. he'd be comfortable reporting to if the roles were reversed. This strikes me as a very good framework. You don't have to be friends with everybody, but you should at least enjoy working with them. And if you don't have that, you should at least deeply respect them. But again, if you don't want to spend a lot of time around people you should trust your instincts about that.

While I'm on this topic of hiring, I want to talk about employee equity. Founders screw this up all the time. I think as a rough estimate, you should aim to give about ten percent of the company to the first ten employees.

They have to earn it over four years anyway, and if they're successful, they're going to contribute way more than that. They're going to increase the value of the company way more than that, and if they don't then they won't be around anyway.

For whatever reason founders are usually very stingy with equity to employees and very generous with equity for investors. I think this is totally backwards. I think this is one of the things founders screw up the most often. Employees will only add more value over time. Investors will usually write the check and then, despite a lot of promises, don't usually do that much. Sometimes they do, but your employees are really the ones that build the company over years and years.

So I believe in fighting with investors to reduce the amount of equity they get and then being as generous as you possibly can with employees. The YC companies that have done this well, the YC companies that have been super generous with their equity to early employees, in general, are the most successful ones that we've funded.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
One thing that founders forget is that after they hire employees, they have to retain them. I'm not going to go into full detail here because we're going to have a lecture on this later, but I do want to talk about it a little bit because founders get this wrong so often. You have to make sure your employees are happy and feel valued. This is one of the reasons that equity grants are so important. People in the excitement of joining a startup don't think about it much, but as they come in day after day, year after year, if they feel they have been treated unfairly that will really start to grate on them and resentment will build.

But more than that, learning just a little bit of management skills, which first-time CEOs are usually terrible at, goes a long way. One of the speakers at YC this summer, who is now extremely successful, struggled early on and had his team turn over a few times. Someone asked him what his biggest struggle was and he said, turns out you shouldn't tell your employees they're fucking up every day unless you want them all to leave because they will.

But as a founder, this is a very natural instinct. You think you can do everything the best and it's easy to tell people when they're not doing it well. So learning just a little bit here will prevent this massive team churn. It also doesn't come naturally to most founders to really praise their team. It took me a little while to learn this too. You have to let your team take credit for all the good stuff that happens, and you take responsibility for the bad stuff.

You have to not micromanage. You have to continually give people small areas of responsibility. These are not the things that founders think about. I think the best thing you can do as a first-time founder is to be aware that you will be a very bad manager and try to overcompensate for that. Dan Pink talks about these three things that motivate people to do great work: autonomy, mastery, and purpose. I never thought about that when I was running my company but I've thought about since and I think that's actually right. I think it's worth trying to think about that. It also took me a while to learn to do things like one on one and to give clear feedback.

All of these things are things first time CEO don't normally do, and maybe I can save you from not doing that.

The last part on the team section is about firing people when it's not working. No matter what I say here is not going to prevent anyone from doing it wrong and the reason that I say that is that firing people is one of the worst parts of running a company. Actually in my own experience, I'd say it is the very worst part. Every first time founder waits too long, everyone hopes that an employee will turn around. But the right answer is to fire fast when it's not working. It's better for the company, it's also better for the employee. But it's so painful and so awful, that everyone gets it wrong the first few times.

In addition to firing people who are doing bad at their job, you also wanna fire people who are a) creating office politics, and b) who are persistently negative. The rest of the company is always aware of employees doing things like this, and it's just this huge drag - it's completely toxic to the company. Again, this is an example of something that might work OK in a big company, although I'm still skeptical, but will kill a startup. So that you need to watch out for people that are ifs.

So, the question is, how do you balance firing people fast and making early employees feel secure? The answer is that when an employee's not working, it's not like they screw up once or twice. Anyone will screw up once or twice, or more times than that, and you know you should be like very loving, not take it out on them, like, be a team, work together.
If someone is getting every decision wrong, that's when you need to act, and at that point it'll be painfully aware to everyone. It's not a case of a few screw-ups, it's a case where every time someone does something, you would have done the opposite yourself. You don't get to make their decisions but you do get to choose the decision-makers. And, if someone's doing everything wrong, just like a consistent thing over like a period of many weeks or a month, you'll be aware of it.

This is one of those cases where in theory, it sounds complicated to be sure what you're talking about, and in practice there's almost never any doubt. It's the difference between someone making one or two mistakes and just constantly screwing everything up, or causing problems, or making everyone unhappy, is painfully obvious the first time you see it.

When should co-founders decide on the equity split?

For some reason, I've never really been sure why this is, a lot of founders, a lot of co-founders like to leave this off for a very long time. You know, they'll even sign the incorporation documents in some crazy way so that they can wait to have this discussion.

This is not a discussion that gets easier with time, you wanna set this ideally very soon after you start working together. And it should be near-equal. If you're not willing to give someone - your co-founder - you know, like an equal share of the equity, I think that should make you think hard about whether or not you want them as a co-founder. But in any case, you should try to have the ink dry on this before the company gets too far along. Like, certainly in the first number of weeks.

So the question is - I said that inexperience is OK - how do you know if someone's gonna scale past, not scale up to a role, as things go on and later become crippling. People that are really smart and that can learn new things can almost always find a role in the company as time goes on. You may have to move them into something else, something other than where they started. You know, it may be that you hire someone to lead the engineering team that over time can't scale as you get up to 50 people, and you give them a different role. Really good people that can almost find some great place in the company, I have not seen that be a problem too often.

So the question is what happens when your relationship with your cofounder falls apart. We're gonna have a session on mechanics later on in the course, but here is the most important thing that founders screw up. Which is, every cofounder, you yourself of course, has to have vesting. Basically what you're doing with cofounder vesting is you're pre-negotiating what happens if one of you leaves. And so the normal stance on this in Silicon Valley is that it takes four years, let's say you split the equity fifty-fifty, is that it takes four years to earn all of that. And the clock doesn't start until one year in. So if you leave after one year, you keep twenty-five percent of the equity, and if you leave after two years, fifty, and on and on like that.

If you don't do that and if you have a huge fallout and one founder leaves early on with half the company, you have this deadweight on your equity table, and it's very hard to get investors to fund you or to do anything else. So number one piece of advice to prevent that is to have vesting on the equity. We pretty much won't fund a company now where the founders don't have vested equity because it's just that hard to do. The other thing that comes up in the relationship between the cofounders, which happens to some degree in every company, is talk about it early, don't let it sit there and fester.

If you have to choose between hiring a sub-optimal employee and losing your customers to a competitor, what do you do? If it's going to be one of the first five employees at a company I would lose those customers. The damage that it does to the company- it's better to lose some customers than
to kill the company. Later on, I might have a slightly different opinion, but it's really hard to say in the general case.

I am going to get to that later. The question is: what about cofounders that aren't working in the same location? The answer is, don't do it. I am skeptical of remote teams in general but in the early days of a startup, when communication and speed outweigh everything else, for some reason video conferencing calls just don't work that well. The data on this is look at say the most dirty successful software companies of all time and try to point to a single example where the cofounders were in different locations. It's really really tough.

Alright, so now we're going to talk about execution. Execution for most founders is not the most fun part of running the company, but it is the most critical. Many cofounders think they're just signing up to this beautiful idea and then they're going to go be on magazine covers and go to parties. But really what it's about more than anything else, what being a cofounder really means, is signing up for this years long grind on execution and you can't outsource this.

The way to have a company that executes well is you have to execute well yourself. Every thing at a startup gets modeled after the founders. Whatever the founders do becomes the culture. So if you want a culture where people work hard, pay attention to detail, manage the customers, are frugal, you have to do it yourself. There is no other way. You cannot hire a COO to do that while you go off to conferences. The company just needs to see you as this maniacal execution machine. As I said in the first lecture, there's at least a hundred times more people with great ideas than people who are willing to put in the effort to execute them well. Ideas by themselves are not worth anything, only executing well is what adds and creates value.

A big part of execution is just putting in the effort, but there is a lot you can learn about how to be good at it. And so we're going to have three classes that just talk about this.

The CEO, people ask me all the time about the jobs of the CEO. There are probably more than five, here are five that come up a lot in the early days. The first four everyone thinks of as CEO jobs: set the vision, raise money, evangelize the mission to people you're trying to recruit, executives, partners, press, everybody, hire and manage the team. But the fifth one is setting the execution bar and this is not the one that most founders get excited about or envision themselves doing but I think it is actually one of the critical CEO roles and no one but the CEO can do this.

Execution gets divided into two key questions. One, can you figure out what to do and two, can you get it done. So I want to talk about two parts of getting it done, assuming that you’ve already figured out what to do. And those are focus and intensity. So focus is critical. One of my favorite questions to ask founders about what they’re spending their time and their money on. This reveals almost everything about what founders think is important.

One of the hardest parts about being a founder is that there are a hundred important things competing for your attention every day. And you have to identify the right two or three, work on those, and then ignore, delegate, or defer the rest. And a lot of these things that founders think are important, interviewing a lot at different law firms, going to conferences, recruiting advisers, whatever, they just don’t matter. What really does matter varies with time, but it’s an important piece of advice. You need to figure out what the one or two most important things are, and then just do those.

And you can only have two or three things every day, because everything else will just come at you. There will be fires every day and if you don’t get good at setting what those two or three things are,
you’ll never be good at getting stuff done. This is really hard for founders. Founders get excited about starting new things.

Unfortunately the trick to great execution is to say no a lot. You’re saying no ninety-seven times out of a hundred, and most founders find they have to make a very conscious effort to do this. Most startups are nowhere near focused enough. They work really hard—maybe—but they don’t work really hard at the right things, so they’ll still fail. One of the great and terrible things about starting a start up is that you get no credit for trying. You only get points when you make something the market wants. So if you work really hard on the wrong things, no one will care.

So then there’s this question of how do you figure out what to focus on each day. Each day it’s really important to have goals. Most good founders I know have a set of small overarching goals for the company that everybody in the company knows. You know it could be something like ship a product by this date, get this certain growth rate, get this engagement rate, hire for these key roles, those are some of them but everyone in the company can tell you each week what are our key goals. And then everybody executes based off of that.

The founders really set the focus. Whatever the founders care about, whatever the founders focus on, that’s going to set the goals for the whole company. The best founders repeat these goals over and over, far more often than they think they should need to. They put them up on the walls they talk about them in one on ones and at all-hands meetings each week. And it keeps the company focus. One of the keys to focus, and why I said cofounders that aren’t friends really struggle, is that you can’t be focused without good communication. Even if you have only four or five people at a company, a small communication breakdown is enough for people to be working on slightly different things. And then you lose focus and the company just scrambles.

I’m going to talk about this a little bit later, but growth and momentum are something you can never lose focus on. Growth and momentum are what a startup lives on and you always have to focus on maintaining these. You should always know how you’re doing against your metrics, you should have a weekly review meeting every week, and you should be extremely suspicious if you’re ever talking about, we’re not focused on growth right now, we’re not growing that well right now but we’re doing this other thing, we don’t have a timeline for when we are going to ship this because we’re focused on this other thing, we’re doing a re-brand, whatever, almost always a disaster.

So you want to have the right metrics and you want to be focused on growing those metrics and having momentum. Don’t let the company get distracted or excited about other things. A common mistake is that companies get excited by their own PR. It’s really easy to get PR with no results and it actually feels like you’re really cool. But in a year you’ll have nothing, and at that point you won’t be cool anymore, and you’ll just be talking about these articles from a year ago that, Oh you know these Stanford students start a new start up, it’s going to be the next big thing and now you have nothing and that sucks.

As I mentioned already, be in the same space. I think this is pretty much a nonstarter. Remote confounding teams is just really really hard. It slows down the cycle time more than anybody ever thinks it’s going to.

The other piece for focus besides execution is intensity. Startups only work at a fairly intense level. A friend of mine says the secret to start up success is extreme focus and extreme dedication. You can have a startup and one other thing, you can have a family, but you probably can’t have many other things. Startups are not the best choice for work life balance and that’s sort of just the sad reality. There’s a lot
of great things about a startup, but this is not one of them. Startups are all-consuming in a way that is generally difficult to explain. You basically need to be willing to outwork your competitors.

The good news here is that a small amount of extra work on the right thing makes a huge difference. One example that I like to give is thinking about the viral coefficient for a consumer web product. How many new users each existing user brings in. If it's .99 the company will eventually flatline and die. And if it's 1.01 you'll be in this happy place of exponential growth forever.

So this is one concrete example of where a tiny extra bit of work is the difference between success and failure. When we talk to successful founders they tell stories like this all the time. Just outworking their competitors by a little bit was what made them successful.

So you have to be really intense. This only comes from the CEO, this only comes from the founders. One of the biggest advantages that start ups have is execution speed and you have to have this relentless operating rhythm. Facebook has this famous poster that says move fast and break things. But at the same time they manage to be obsessed with quality. And this is why it's hard. It's easy to move fast or be obsessed with quality, but the trick is to do both at a startup. You need to have a culture where the company has really high standards for everything everyone does, but you still move quickly.

Apple, Google, and Facebook have each done this extremely well. It's not about the product, it's about everything they do. They move fast and they break things, they're frugal in the right places, but they care about quality everywhere. You don't buy people shitty computers if you don't want them to write shitty code. You have to set a quality bar that runs through the entire company. Related to this is that you have to be decisive. Indecisiveness is a startup killer. Mediocre founders spend a lot of time talking about grand plans, but they never make a decision. They're talking about you know I could do this thing, or I could do that other thing, and they're going back and forth and they never act. And what you actually need is this bias towards action.

The best founders work on things that seem small but they move really quickly. But they get things done really quickly. Every time you talk to the best founders they've gotten new things done. In fact, this is the one thing that we learned best predicts a success of founders in YC. If every time we talk to a team they've gotten new things done, that's the best predictor we have that a company will be successful. Part of this is that you can do huge things in incremental pieces. If you keep knocking down small chunks one at a time, in a year you look back and you've done this amazing thing. On the other hand, if you disappear for a year and you expect to come back with something amazing all at once, it usually never happens.

So you have to pick these right size projects. Even if you're building this crazy biosynthetic company and you say well I have to go away for a year, there's no way to do this incrementally, you can still usually break it into smaller projects.

So speed is this huge premium. The best founders usually respond to e-mail the most quickly, make decisions most quickly, they're generally quick in all of these ways. And they had this do what ever it takes attitude.

They also show up a lot.

They come to meetings, they come in, they meet us in person. One piece of advice that I have that's always worked for me: they get on planes in marginal situations. I'll tell a quick story here.
When I was running my own company, we found out we were about to lose a deal. It was sort of this critical deal from the first big customer in the space. And it was going to go to this company that had been around for year before we were. And they had this like all locked up. And we called and said “we have this better product you have to meet with us” and they said “well we’re signing this deal tomorrow. sorry.” We drove to the airport, we got on a plane, we were at their office at 6am the next morning. We just sat there, they told us to go away, we just kept sitting there. Finally once of the junior guys decided to meet with us, after that, finally one of the senior guys decided to meet with us. They ended up ripping up the contract with the other company, and we closed the deal with them about a week later. And I’m sure, that had we not gotten on a plane, had we not shown up in person, that would not have worked out.

And so, you just sort of show and and do these things, when people say get on plane in marginal situations, they actually mean it, but they don’t mean it literally. But I actually think it’s good, literal advice.

So I mentioned this momentum and growth earlier. Once more: the momentum and growth are the lifeblood of startups. This is probably in the top three secrets of executing well. You want a company to be winning all the time. If you ever take your foot off the gas pedal, things will spiral out of control, snowball downwards. A winning team feels good and keeps winning. A team that hasn’t won in a while gets demotivated and keeps losing. So always keep momentum, it’s this prime directive for managing a startup. If I can only tell founders one thing about how to run a company, it would be this.

For most software startups, this translates to keep growing. For hardware startups it translates to: don’t let your ship dates slip. This is what we tell people during YC, and they usually listen and everything is good. What happens at the end of YC is that they get distracted on other things, and then growth slows down. And somehow, after that happens, people start getting unhappy and quitting and everything falls apart. It’s hard to figure out a growth engine because most companies grow in new ways, but there’s this thing: if you build a good product it will grow. So getting this product right at the beginning is the best way not to lose momentum later.

If you do lose momentum, most founders try to get it back in the wrong way. They give these long speeches about vision for the company and try to rally the troops with speeches. But employees in a company where momentum has sagged, don’t want to hear that. You have to save the vision speeches for when the company is winning. When you’re not winning, you just have to get momentum back in small wins. A board member of mine used to say that sales fix everything in a startup. And that is really true. So you figure out where you can get these small wins and you get that done. And then you’ll be amazed at how all the other problems in a startup disappear.

Another thing that you’ll notice if you have momentum sag, is that everyone starts disagreeing about what to do. Fights come out when a company loses momentum. And so a framework for that that I think works is that when there’s disagreement among the team about what to do, then you ask your users and you do whatever your users tell you. And you have to remind people: “hey, stuff’s not working right now we don’t actually hate each other, we just need to get back on track and everything will work.” If you just call it out, if you just acknowledge that, you’ll find that things get way better.

To use a Facebook example again, when Facebook’s growth slowed in 2008, mark instituted a “growth group.” They worked on very small things to make Facebook grow faster. All of these by themselves seemed really small, but they got the curve of Facebook back up. It quickly became the most prestigious group there. Mark has said that it’s been one of Facebook’s best innovations. According to friends of mine that worked at Facebook at the time, it really turned around the dynamic of the company. And it
went from this thing where everyone was feeling bad, and momentum was gone, back to a place that was winning.

So a good way to keep momentum is to establish an operating rhythm at the company early. Where you ship product and launch new features on a regular basis. Where you’re reviewing metrics every week with the entire company. This is actually one of the best things your board can do for you. Boards add value to business strategy only rarely. But very frequently you can use them as a forcing function to get the company to care about metrics and milestones.

One thing that often disrupts momentum and really shouldn’t is competitors. Competitors making noise in the press I think probably crushes a company’s momentum more often than any other external factor.

So here’s a good rule of thumb: don’t worry about a competitor at all, until they’re actually beating you with a real, shipped product. Press releases are easier to write than code, and that is still easier than making a great product. So remind your company of this, and this is sort of a founder’s role, is not to let the company get down because of the competitors in the press.

This great quote from Henry Ford that I love: “The competitor to be feared is one who never bothers about you at all, but goes on making his own business better all the time.”

These are almost never the companies that put out a lot of press releases. And they bum people out.
An article came out today in Businessweek about arrogance and Silicon Valley. I thought it was good, but there was one more point I wanted to make.

People often accuse people in Silicon Valley of working on things that don’t matter. Often they're right. But many very important things start out looking as if they don’t matter, and so it’s a very bad mistake to dismiss everything that looks trivial.

The problem comes when people building something claim it's going to change the world when it still looks like a toy. That just pisses people off.

Facebook, Twitter, reddit, the Internet itself, the iPhone, and on and on and on—most people dismissed these things as incremental or trivial when they first came out.

I have a thought about why. There’s the famous observation that the value of a network grows as a function of the square of the number of nodes, and also many of these services/products double their userbase every N months, with N decreasing as the service gets for valuable. So the value/importance of the service grows hyperexponentially. I’ve never met anyone in my life that has a good intuition for hyperexponential growth—most of us even struggle to comprehend exponential growth.

There is all sorts of emergent behavior as something grows in importance a millionfold in a short period of time. If some users really love what you’re building, engage with the service or product as an important part of their daily lives, and interesting new behaviors keep emerging as you grow, keep working on it.

As an aside, pay no attention to market predictions—some of the worst predictions in the history of business (a market for 5 computers, a market for 900,000 cell phones) have been the most costly.

There are two time-tested strategies to change the world with technology. One is to build something that some people love but most people think is a toy; the other is to be hyperambitious and start an electric car company or a rocket company. Most of the “intermediate” companies, although it would take a separate long post to explain why, end up not having a big impact.

In closing, I have two pieces of advice for the “arrogant fucks” who make the world go round. One, don’t claim you’re changing the world until you’ve changed it. Two, ignore the haters and work on whatever you find interesting. The
Lecture 2: Do Things That Don't Scale

http://tech.genius.com/Paul-graham-do-things-that-dont-scale-annotated

Paul Graham

One of the most common types of advice we give at Y Combinator is to do things that don't scale. A lot of would-be founders believe that startups either take off or don't. You build something, make it available, and if you've made a better mousetrap, people beat a path to your door as promised. Or they don't, in which case the market must not exist. [1]

Actually startups take off because the founders make them take off. There may be a handful that just grew by themselves, but usually it takes some sort of push to get them going. A good metaphor would be the cranks that car engines had before they got electric starters. Once the engine was going, it would keep going, but there was a separate and laborious process to get it going.

Recruit

The most common unscaleable thing founders have to do at the start is to recruit users manually. Nearly all startups have to. You can't wait for users to come to you. You have to go out and get them.

Stripe is one of the most successful startups we've funded, and the problem they solved was an urgent one. If anyone could have sat back and waited for users, it was Stripe. But in fact they're famous within YC for aggressive early user acquisition.

Startups building things for other startups have a big pool of potential users in the other companies we've funded, and none took better advantage of it than Stripe. At YC we use the term "Collison installation" for the technique they invented. More diffident founders ask "Will you try our beta?" and if the answer is yes, they say "Great, we'll send you a link." But the Collison brothers weren't going to wait. When anyone agreed to try Stripe they'd say "Right then, give me your laptop" and set them up on the spot.

There are two reasons founders resist going out and recruiting users individually. One is a combination of shyness and laziness. They'd rather sit at home writing code than go out and talk to a bunch of strangers and probably be rejected by most of them. But for a startup to succeed, at least one founder (usually the CEO) will have to spend a lot of time on sales and marketing. [2]

The other reason founders ignore this path is that the absolute numbers seem so small at first. This can't be how the big, famous startups got started, they think. The mistake they make is to underestimate the power of compound growth. We encourage every startup to measure their progress by weekly growth rate. If you have 100 users, you need to get 10 more next week to grow 10% a week. And while 110 may not seem much better than 100, if you keep growing at 10% a week you'll be surprised how big the numbers get. After a year you'll have 14,000 users, and after 2 years you'll have 2 million.

You'll be doing different things when you're acquiring users a thousand at a time, and growth has to slow down eventually. But if the market exists you can usually start by recruiting users manually and then gradually switch to less manual methods. [3]

Airbnb is a classic example of this technique. Marketplaces are so hard to get rolling that you should expect to take heroic measures at first. In Airbnb's case, these consisted of going door to door in New York, recruiting new users and helping existing ones improve their listings. When I remember the Airbnbs during YC, I picture them with rolly bags, because when they showed up for Tuesday dinners they'd always just flown back from somewhere.

Fragile

Airbnb now seems like an unstoppable juggernaut, but early on it was so fragile that about 30 days of going out and engaging in person with users made the difference between success and failure.

That initial fragility was not a unique feature of Airbnb. Almost all startups are fragile initially. And that's one of the biggest things inexperienced founders and investors (and reporters and know-it-alls on forums) get wrong about them. They unconsciously judge larval startups by the standards of established ones. They're like someone looking at a newborn baby and concluding "there's no way this tiny creature could ever accomplish anything."

It's harmless if reporters and know-it-alls dismiss your startup. They always get things wrong. It's even ok if investors dismiss your startup; they'll change their minds when they see growth. The big danger is that you'll dismiss your startup yourself. I've seen it happen. I often have to encourage founders who don't see the full potential of what they're building. Even Bill Gates made that mistake. He returned to Harvard for the fall semester after starting Microsoft. He didn't stay long, but he wouldn't have returned at all if he'd realized Microsoft was going to be even a fraction of the size it turned out to be. [4]

The question to ask about an early stage startup is not "is this company taking over the world?" but "how big could this company get if the founders did the right things?" And the right things often seem both laborious and inconsequential at the time. Microsoft can't have seemed very impressive when it was just a couple guys in Albuquerque writing Basic interpreters for a market of a few thousand hobbyists (as they were then called), but in retrospect that was the optimal path to dominating microcomputer software. And I know
Brian Chesky and Joe Gebbia didn’t feel like they were en route to the big time as they were taking "professional" photos of their first hosts’ apartments. They were just trying to survive. But in retrospect that too was the optimal path to dominating a big market.

How do you find users to recruit manually? If you build something to solve your own problems, then you only have to find your peers, which is usually straightforward. Otherwise you’ll have to make a more deliberate effort to locate the most promising vein of users. The usual way to do that is to get some initial set of users by doing a comparatively untargeted launch, and then to observe which kind seem most enthusiastic, and seek out more like them. For example, Ben Silbermann noticed that a lot of the earliest Pinterest users were interested in design, so he went to a conference of design bloggers to recruit users, and that worked well. [5]

**Delight**

You should take extraordinary measures not just to acquire users, but also to make them happy. For as long as they could (which turned out to be surprisingly long), Wufoo sent each new user a hand-written thank you note. Your first users should feel that signing up with you was one of the best choices they ever made. And you in turn should be racking your brains to think of new ways to delight them.

Why do we have to teach startups this? Why is it counterintuitive for founders? Three reasons, I think.

One is that a lot of of startup founders are trained as engineers, and customer service is not part of the training of engineers. You’re supposed to build things that are robust and elegant, not be slavishly attentive to individual users like some kind of salesperson. Ironically, part of the reason engineering is traditionally averse to handholding is that its traditions date from a time when engineers were less powerful—when they were only in charge of their narrow domain of building things, rather than running the whole show. You can be ornery when you’re Scotty, but not when you’re Kirk.

Another reason founders don’t focus enough on individual customers is that they worry it won’t scale. But when founders of larval startups worry about this, I point out that in their current state they have nothing to lose. Maybe if they go out of their way to make existing users super happy, they’ll one day have too many to do so much for. That would be a great problem to have. See if you can make it happen. And incidentally, when it does, you’ll find that delighting customers scales better than you expected. Partly because you can usually find ways to make anything scale more than you would have predicted, and partly because delighting customers will by then have permeated your culture.

I have never once seen a startup lured down a blind alley by trying too hard to make their initial users happy.

But perhaps the biggest thing preventing founders from realizing how attentive they could be to their users is that they’ve never experienced such attention themselves. Their standards for customer service have been set by the companies they’ve been customers of, which are mostly big ones. Tim Cook doesn’t send you a hand-written note after you buy a laptop. He can’t. But you can. That’s one advantage of being small: you can provide a level of service no big company can. [6]

Once you realize that existing conventions are not the upper bound on user experience, it’s interesting in a very pleasant way to think about how far you could go to delight your users.

**Experience**

I was trying to think of a phrase to convey how extreme your attention to users should be, and I realized Steve Jobs had already done it: insanely great. Steve wasn't just using "insanely" as a synonym for "very." He meant it more literally—that one should focus on quality of execution to a degree that in everyday life would be considered pathological.

All the most successful startups we've funded have, and that probably doesn't surprise would-be founders. What novice founders don’t get is what insanely great translates to in a larval startup. When Steve Jobs started using that phrase, Apple was already an established company. He meant the Mac (and its documentation and even packaging—such is the nature of obsession) should be insanely well designed and manufactured. That's not hard for engineers to grasp. It's just a more extreme version of designing a robust and elegant product.

What founders have a hard time grasping (and Steve himself might have had a hard time grasping) is what insanely great morphs into as you roll the time slider back to the first couple months of a startup's life. It's not the product that should be insanely great, but the experience of being your user. The product is just one component of that. For a big company it's necessarily the dominant one. But you can and should give users an insanely great experience with an early, incomplete, buggy product, if you make up the difference with attentiveness.

Can, perhaps, but should? Yes. Over-engaging with early users is not just a permissible technique for getting growth rolling. For most successful startups it's a necessary part of the feedback loop that makes the product good. Making a better mousetrap is not an atomic operation. Even if you start the way most successful startups have, by building something you yourself need, the first thing you build is never quite right. And except in domains with big penalties for making mistakes, it's often better not to aim for perfection initially. In software, especially, it usually works best to get something in front of users as soon as it has a quantum of utility, and then see what they do with it. Perfectionism is
often an excuse for procrastination, and in any case your initial model of users is always inaccurate, even if you're one of them. [7]

The feedback you get from engaging directly with your earliest users will be the best you ever get. When you're so big you have to resort to focus groups, you'll wish you could go over to your users' homes and offices and watch them use your stuff like you did when there were only a handful of them.

**Fire**

Sometimes the right unscalable trick is to focus on a deliberately narrow market. It's like keeping a fire contained at first to get it really hot before adding more logs.

That's what Facebook did. At first it was just for Harvard students. In that form it only had a potential market of a few thousand people, but because they felt it was really for them, a critical mass of them signed up. After Facebook stopped being for Harvard students, it remained for students at specific colleges for quite a while. When I interviewed Mark Zuckerberg at Startup School, he said that while it was a lot of work creating course lists for each school, doing that made students feel the site was their natural home.

Any startup that could be described as a marketplace usually has to start in a subset of the market, but this can work for other startups as well. It's always worth asking if there's a subset of the market in which you can get a critical mass of users quickly. [8]

Most startups that use the contained fire strategy do it unconsciously. They build something for themselves and their friends, who happen to be the early adopters, and only realize later that they could offer it to a broader market. The strategy works just as well if you do it unconsciously. The biggest danger of not being consciously aware of this pattern is for those who naively discard part of it. E.g. if you don't build something for yourself and your friends, or even if you do, but you come from the corporate world and your friends are not early adopters, you'll no longer have a perfect initial market handed to you on a platter.

Among companies, the best early adopters are usually other startups. They're more open to new things both by nature and because, having just been started, they haven't made all their choices yet. Plus when they succeed they grow fast, and you with them. It was one of many unforeseen advantages of the YC model (and specifically of making YC big) that B2B startups now have an instant market of hundreds of other startups ready at hand.

**Meraki**

For hardware startups there's a variant of doing things that don't scale that we call "pulling a Meraki." Although we didn't fund Meraki, the founders were Robert Morris's grad students, so we know their history. They got started by doing something that really doesn't scale: assembling their routers themselves.

Hardware startups face an obstacle that software startups don't. The minimum order for a factory production run is usually several hundred thousand dollars. Which can put you in a catch-22: without a product you can't generate the growth you need to raise the money to manufacture your product. Back when hardware startups had to rely on investors for money, you had to be pretty convincing to overcome this. The arrival of crowdfunding (or more precisely, preorders) has helped a lot. But even so I'd advise startups to pull a Meraki initially if they can. That's what Pebble did. The Pebbles assembled the first several hundred watches themselves. If they hadn't gone through that phase, they probably wouldn't have sold $10 million worth of watches when they did go on Kickstarter.

Like paying excessive attention to early customers, fabricating things yourself turns out to be valuable for hardware startups. You can tweak the design faster when you're the factory, and you learn things you'd never have known otherwise. Eric Migicovsky of Pebble said one of things he learned was "how valuable it was to source good screws." Who knew?

**Consult**

Sometimes we advise founders of B2B startups to take over-engagement to an extreme, and to pick a single user and act as if they were consultants building something just for that one user. The initial user serves as the form for your mold; keep tweaking till you fit their needs perfectly, and you'll usually find you've made something other users want too. Even if there aren't many of them, there are probably adjacent territories that have more. As long as you can find just one user who really needs something and can act on that need, you've got a toehold in making something people want, and that's as much as any startup needs initially. [9]

Consulting is the canonical example of work that doesn't scale. But (like other ways of bestowing one's favors liberally) it's safe to do it so long as you're not being paid to. That's where companies cross the line. So long as you're a product company that's merely being extra attentive to a customer, they're very grateful even if you don't solve all their problems. But when they start paying you specifically for that attentiveness—when they start paying you by the hour—they expect you to do everything.

Another consulting-like technique for recruiting initially lukewarm users is to use your software yourselves on their behalf. We did that at Viaweb. When we approached merchants asking if they wanted to use our software to make online stores, some said no, but they'd let us make one for them. Since we would do anything to get users, we did. We felt pretty lame at the time. Instead of organizing big strategic e-commerce partnerships, we were trying to sell luggage and pens and men's shirts. But in retrospect it was exactly the right thing to do, because it taught us how it
would feel to merchants to use our software. Sometimes the feedback loop was near instantaneous: in the middle of building some merchant's site I'd find I needed a feature we didn't have, so I'd spend a couple hours implementing it and then resume building the site.

**Manual**

There's a more extreme variant where you don't just use your software, but are your software. When you only have a small number of users, you can sometimes get away with doing by hand things that you plan to automate later. This lets you launch faster, and when you finally automate yourself out of the loop, you'll know exactly what to build because you'll have muscle memory from doing it yourself.

When manual components look to the user like software, this technique starts to have aspects of a practical joke. For example, the way Stripe delivered “instant” merchant accounts to its first users was that the founders manually signed them up for traditional merchant accounts behind the scenes.

Some startups could be entirely manual at first. If you can find someone with a problem that needs solving and you can solve it manually, go ahead and do that for as long as you can, and then gradually automate the bottlenecks. It would be a little frightening to be solving users' problems in a way that wasn't yet automatic, but less frightening than the far more common case of having something automatic that doesn't yet solve anyone's problems.

**Big**

I should mention one sort of initial tactic that usually doesn't work: the Big Launch. I occasionally meet founders who seem to believe startups are projectiles rather than powered aircraft, and that they'll make it big if and only if they're launched with sufficient initial velocity. They want to launch simultaneously in 8 different publications, with embargoes. And on a tuesday, of course, since they read somewhere that's the optimum day to launch something.

It's easy to see how little launches matter. Think of some successful startups. How many of their launches do you remember? All you need from a launch is some initial core of users. How well you're doing a few months later will depend more on how happy you made those users than how many there were of them.[10]

So why do founders think launches matter? A combination of solipsism and laziness. They think what they're building is so great that everyone who hears about it will immediately sign up. Plus it would be so much less work if you could get users merely by broadcasting your existence, rather than recruiting them one at a time. But even if what you're building really is great, getting users will always be a gradual process—partly because great things are usually also novel, but mainly because users have other things to think about.

Partnerships too usually don't work. They don't work for startups in general, but they especially don't work as a way to get growth started. It's a common mistake among inexperienced founders to believe that a partnership with a big company will be their big break. Six months later they're all saying the same thing: that was way more work than we expected, and we ended up getting practically nothing out of it. [11]

It's not enough just to do something extraordinary initially. You have to make an extraordinary effort initially. Any strategy that omits the effort—whether it's expecting a big launch to get you users, or a big partner—is ipso facto suspect.

**Vector**

The need to do something unscalably laborious to get started is so nearly universal that it might be a good idea to stop thinking of startup ideas as scalars. Instead we should try thinking of them as pairs of what you're going to build, plus the unscalable thing(s) you're going to do initially to get the company going.

It could be interesting to start viewing startup ideas this way, because now that there are two components you can try to be imaginative about the second as well as the first. But in most cases the second component will be what it usually is—recruit users manually and give them an overwhelmingly good experience—and the main benefit of treating startups as vectors will be to remind founders they need to work hard in two dimensions. [12]

In the best case, both components of the vector contribute to your company's DNA: the unscalable things you have to do to get started are not merely a necessary evil, but change the company permanently for the better. If you have to be aggressive about user acquisition when you're small, you'll probably still be aggressive when you're big. If you have to manufacture your own hardware, or use your software on users's behalf, you'll learn things you couldn't have learned otherwise. And most importantly, if you have to work hard to delight users when you only have a handful of them, you'll keep doing it when you have a lot.

**Notes**

[1] Actually Emerson never mentioned mousetraps specifically. He wrote "If a man has good corn or wood, or boards, or pigs, to sell, or can make better chairs or knives, crucibles or church organs, than anybody else, you will find a broad hard-beaten road to his house, though it be in the woods."

[2] Thanks to Sam Altman for suggesting I make this explicit. And no, you can't avoid doing sales by hiring someone to do it for you. You have to do sales yourself initially. Later you
can hire a real salesperson to replace you.

[3] The reason this works is that as you get bigger, your size helps you grow. Patrick Collison wrote "At some point, there was a very noticeable change in how Stripe felt. It tipped from being this boulder we had to push to being a train car that in fact had its own momentum."

[4] One of the more subtle ways in which YC can help founders is by calibrating their ambitions, because we know exactly how a lot of successful startups looked when they were just getting started.

[5] If you're building something for which you can't easily get a small set of users to observe—e.g. enterprise software—and in a domain where you have no connections, you'll have to rely on cold calls and introductions. But should you even be working on such an idea?

[6] Garry Tan pointed out an interesting trap founders fall into in the beginning. They want so much to seem big that they imitate even the flaws of big companies, like indifference to individual users. This seems to them more "professional." Actually it's better to embrace the fact that you're small and use whatever advantages that brings.

[7] Your user model almost couldn't be perfectly accurate, because users' needs often change in response to what you build for them. Build them a microcomputer, and suddenly they need to run spreadsheets on it, because the arrival of your new microcomputer causes someone to invent the spreadsheet.

[8] If you have to choose between the subset that will sign up quickest and those that will pay the most, it's usually best to pick the former, because those are probably the early adopters. They'll have a better influence on your product, and they won't make you expend as much effort on sales. And though they have less money, you don't need that much to maintain your target growth rate early on.

[9] Yes, I can imagine cases where you could end up making something that was really only useful for one user. But those are usually obvious, even to inexperienced founders. So if it's not obvious you'd be making something for a market of one, don't worry about that danger.

[10] There may even be an inverse correlation between launch magnitude and success. The only launches I remember are famous flops like the Segway and Google Wave. Wave is a particularly alarming example, because I think it was actually a great idea that was killed partly by its overdone launch.

[11] Google grew big on the back of Yahoo, but that wasn't a partnership. Yahoo was their customer.

[12] It will also remind founders that an idea where the second component is empty—an idea where there is nothing you can do to get going, e.g. because you have no way to find users to recruit manually—is probably a bad idea, at least for those founders.

Thanks to Sam Altman, Paul Buchheit, Patrick Collison, Kevin Hale, Steven Levy, Jessica Livingston, Geoff Ralston, and Garry Tan for reading drafts of this.
3. How to have Ideas
Lecture 3: Counterintuitive Parts of Startups, and How to Have Ideas

http://startupclass.samaltman.com/courses/lec03/

Paul Graham

One of the advantages of having kids is that when you have to give advice to people you can ask yourself, what would I tell my own kids, and actually you'll find this really focuses you. So even though my kids are little, my two year old today, one asked what he'll be after two said, a bat. If the correct answer was three, but a bat is so much more interesting. So even though my kids are little I already know what I would tell them about startups, if they were in college, so that is what I'm going to tell you; so you're literary going to get what I would tell my own kids, since most of you are young enough to be my own kids.

Startups are very counterintuitive and I'm not sure exactly why, it could be simply be because knowledge about them has not permitted our culture yet, but whatever the reason this is an area where you cannot trust your intuition all the time. It's like skiing in that way, any of you guys learn to ski as adults? When you first try skiing and you want to slow down, your first impulse is to lean back, just like everything else. But lean back on the skis and you fly down the hill out of control, so as I learned, part of learning to ski is learning to suppress that impulse. Eventually you get new habits, but in the begging there is this list of things you're trying to remember as you start down the hill, like, alternate feet, make s-turns, do not drag the inside foot.

Startups are as unnatural as skiing and there is a similar list of stuff you have to remember for startups. What I'm going to give you today is the beginning of the list, the list of the counterintuitive stuff you have to remember, to prevent your existing instincts from leading you astray.

The first thing on it is the fact I just mentioned: startups are so weird that if you follow your instincts they will lead you astray. If you remember nothing more than that, when you're about to make a mistake, you can pause before making it. When I was running Y Combinator we used to joke that our function was to tell founders things they would ignore, and it's really true; Batch after batch the YC partners warned founders about mistakes they were about to make and the founders ignored them, and they came back a year later and said, "I wished we'd listened, " but that dude is in their cap table and there is nothing they can do.

Q: Why do founders persistently ignore the partner’s advice?

A: That's the thing about counterintuitive ideas, they contradict your intuitions, they seem wrong, so, of course, your first impulse is to ignore them and, in fact, that's not just the curse of Y Combinator, but to some extent our raison d'etre. You don't need people to give you advice that does not surprise you; If founders existing intuition gave them the right answers, they would not need us. That's why there is a lot of ski instructors, and not many running instructors; you don't see those words together, running instructor, as much as you see, ski instructor. It's because skiing is counterintuitive, sort of what YC is, business ski instructors, except you are going up slopes instead of down them, well ideally.

You can, however, trust your instincts about people. Your life so far hasn't been much like starting a startup, but all the interactions you've had with people are just like the interactions you have with people in the business world. In fact, one of the big mistakes that founders make is to not trust their intuition about people enough. They meet someone, who seems impressive, but about whom they feel
some misgivings and then later when things blow up, they say, "You know I knew there was something wrong about that guy, but I ignored it because he seemed so impressive."

There is this specific sub-case in business, especially if you come from an engineering background, as I believe you all do. You think business is supposed to be this slightly distasteful thing. So when you meet people who seem smart, but somehow distasteful, you think, "Okay this must be normal for business," but it's not. Just pick people the way you would pick people if you were picking friends. This is one of those rare cases where it works to be self indulgent. Work with people you would generally like and respect, and that you have known long enough to be sure about because there are a lot of people who are really good at seeming likable for a while. Just wait till your interests are imposed and then you'll see.

The second counterintuitive point, this might come as a little bit of a disappointment, but what you need to succeed in a startup is not expertise in startups. That makes this class different from most other classes you take. You take a French class, at end of it you learn how to speech French. You do the work, you may not sound exactly like a French person, but pretty close, right? This class can teach you about start-ups, but that is not what you need to know. What you need to know to succeed in a startup is not expertise in startups, what you need is expertise in your own users.

Mark Zuckerberg did not succeed at Facebook because he was an expert in start-ups, he succeeded despite being a complete noob at startups; I mean Facebook was first incorporated as a Florida LLC. Even you guys know better than that. He succeeded despite being a complete noob at startups because he understood his users very well. Most of you don't know the mechanics of raising an angel round, right? If you feel bad about that, don't, because I can tell you Mark Zuckerberg probably doesn't know the mechanics of raising an angel round either; if he was even paying attention when Ron Conway wrote him the big check, he probably has forgotten about it by now.

In fact, I worry it's not merely unnecessary for people to learn in detail about the mechanics of starting a startup, but possibly somewhat dangerous because another characteristic mistake of young founders starting startups, is to go through the motions of starting a startup. They come up with some plausible sounding idea, they raise founding to get a nice evaluation, they rent a nice office in Soma, hire a bunch of their friends; then the next step after they rent a nice office in Soma and hire a bunch of their friends, is they gradually realize how completely fucked they are because while imitating all the outward forms of starting a startup, they have neglected the one thing that is actually essential, which is to make something people want. By the way that's the only use of that swear word, except for the initial one, that was involuntary and I did check with Sam if it would be okay; he said he had done it several times, I mean use the word.

We saw this happen so often, no I mean people going through the emotions of starting a startup that we made up a name for it: "Playing House." Eventually I realized why it was happening, the reason young founders go though the motions of starting a star-up is because that is what they have been trained to do, their whole life, up to this point. Think about what it takes to get into college: extracurricular activities? Check. Even in college classes most of the work you do is as artificial as running laps, and I'm not attacking the educational system for being this way, inevitably the work that you do to learn something is going to have some amount of fakeness to it. And if you measure people's performance they will inevitably exploit the difference to the degree that what you're measuring is largely an artifact of the fakeness.

I confess that I did this myself in college; in fact, here is a useful tip on getting good grades. I found that in a lot of classes there might only be twenty or thirty ideas that had the right shape to make good exam questions. So the way I studied for exams in these classes was not to master the material in the
class, but to try and figure out what the exam questions would be and work out the answer in advance. For me the test was not like, what my answers would be on my exam, for me the test was which of my exam questions would show up on the exam. So I would get my grade instantly, I would walk into the exam and look at the questions and see how many I got right, essentially. It works in a lot of classes, especially CS classes. I remember automata theory, there is only a few things that make since to ask about automata theory.

So it's not surprising that after being effetely trained for their whole lives to play such games; young founders' first impulse on starting a startup is to find out what the tricks are for this new game, what are the extracurricular activities of startups, what are things I have to do? They always want to know, since apparently the measure of success for a startup is fundraising, another noob mistake. They always want to know, what are the tricks for convincing investors? And we have to tell them the best way to convince investors is to start a startup that is actually doing well, meaning growing fast, and then simply tell investors so.

Then they ask okay, so what are the tricks for growing fast, and this is exacerbated by the existence of this term, "Growth Hacks." Whenever you hear somebody talk about Growth Hacks, just mentally translate it in your mind to, bullshit, because what we tell them is the way to make your startup grow is to make something that users really love, and then tell them about it. So that's what you have to do, that's Growth Hacks right there.

So many of the conversations the YC partners have with the founders begin with the founders saying a sentence that begins with, "How do I, " and the partners answering with a sentence that begins with, "Just." Why do they make things so complicated? The reason, I realized, after years of being puzzled by this, is they're looking for the trick, they've been trained to look 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 may be able to succeed by sucking up to the right person; Giving the impression of productivity by sending emails late at night, or if you're smart enough changing the clock on your computer, cause who's going to check the headers, right? I like an audience I can tell jokes to and they laugh. Over in the business school: headers? Okay, God this thing is being recorded, I just realized that.

Alright for now on we are sticking strictly to the script. But, in startups, that does not work. There is no boss to trick, how can you trick people, when there is nobody to trick? There are only users and all users care about is whether your software does what they want, right? They're like sharks, sharks are too stupid to fool, you can't wave a red flag and fool it, it's like meat or no meat. You have to have what people want and you only prosper to the extent that you do. The dangerous thing is, especially for you guys, the dangerous thing is that faking does work to some extent with investors.

If you’re really good at knowing what you’re talking about, you can fool investors, for one, maybe two rounds of funding, but it’s not in your interest to do. I mean, you’re all doing this for equity, you’re pulling a confidence trick on yourself. Wasting your own time, because the startup is doomed and all you’re doing is wasting your time writing 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. Someone who knows zero about fundraising, but has made something users really love, will have an easier time raising money than someone who knows every trick in the book, but has a flat usage graph.
Though, in a sense, it's bad news that gaming the startups stops work, gaming the system stops working now. In the sense that you're deprived of your most powerful weapons and, after all, you spent twenty years mastering. It is something, I find it very exciting, that there even exist parts of the world where gaming the system is not how you win. I would have been really excited in college if I explicitly realized that there are parts of the world where gaming the system matters less than others, and some where it hardly matters at all, but there are and this is one of the most important thing to think about when planning your future. How do you win at each type of work, and what do you want to win by doing?

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 that you cannot imagine and if it succeeds it will take over your life for a long time; for several years, at the very least, maybe a decade, maybe the rest of your working life. So there is a real opportunity cost here. It may seem to you that Larry Page has an enviable life, but there are parts of it that are defiantly unenviable. The way the world looks to him is that he started running as fast as he could, at age twenty-five, and he has not stopped to catch his breath since. Every day shit happens within the Google empire that only the emperor can deal with and he, as the emperor, has to deal with it. If he goes on vacation for even a week, a whole backlog of shit accumulates, and he has to bear this, uncomplaining, because number one as the company's daddy he cannot show fear or weakness. And number two, if you're a billionaire, you get zero, actually less than zero sympathy, if you complain about having a difficult life.

Which has this strange side effect that the difficulty of being a successful startup founder is concealed from almost everyone who has done it. People who win the one-hundred meters in the Olympics, they walk up to them and they're [out of breath]. Larry Page is doing that too, but you never get to see it.

Y Combinator has now funded several companies that could be called big successes and in every single case the founder says the same thing, "It never gets any easier." The nature of the problems change, so you maybe worrying about more glamorous problems like, construction delays in your new London offices, rather than 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, and it's like a button you press and it changes your life irrevocably. While it's honestly the best thing having kids, if you take away one thing from this lecture, remember that. There are a lot of things that are easier to do before you have kids than after may of which will make you a better parent, when you do have kids, and so, in rich countries, most people delay pushing the button for a while and I'm sure you are all intimately familiar with that procedure.

Yet when it comes to starting startups a lot of people seem to think they are supposed to start them in college, are you crazy? And what are the universities thinking, they go out of their way to ensure that their students are well supplied with contraceptives, and yet they are starting 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, university's are at least de-facto supposed to prepare you for your career, and so if you're interested in startups, it seems like universities are supposed to teach you about startups and if they don't maybe they loose applicants to universities that do claim to do that. So can universities teach you about startups? Well, if not, what are we doing here? Yes and no, as I've explained to you about start-ups, but this is not what you need to know. Essentially, if you want to learn French, universities can teach you linguistics, that is what this is. This is linguistics we're teaching you how to learn languages and what you need to know is how a particular language.
What you need to know are the needs of your own users. You can't learn those until you actually start the company, which means that starting a startup is something you can intrinsically only learn by doing it. You can't do that in college for the reason I just explained. Startups take over your entire life. If you start a startup in college, if you start a startup as a student, you can't start a startup as a student because if you start a startup you're not a student anymore. You may be normally a student but you won't even be that for very much longer. 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. Well, I can answer that one for you. I'm talking to my own kids here. Do not start a startup in college. I hope I'm not disappointing anyone seriously. Starting a startup could be a good component of a good life for a lot of ambitious people. This is just a part of a much bigger problem that you are trying to solve. How to have a good life, right. Those that are starting a startup could be a good thing to do at some point. Twenty is not the optimal time to do it.

There are things that you can do in your early twenties that you cannot do as well before or after. Like plunge deeply into projects on a whim that seem like they will have no pay off. Travel super cheaply with no sense of a deadline. In fact they are really metamorphic shapes in different domains.

For unambitious people your thing can be the dreaded failure to launch. For the ambitious ones it's a really valuable sort of exploration and if you start a startup at twenty and you are sufficiently successful you will never get to do it.

Mark Zuckerberg will never get to bum around a foreign country. If he goes to a foreign county, it's either as a de-facto state visit or like he's hiding out incognito at George V in Paris. He's never going to just like backpack around Thailand if that's still what people do. Do people still backpack around Thailand? That's the first real enthusiasm I've ever seen from this class. Should have given this talk in Thailand. He can do things you can't do, like charter jets to fly him to foreign countries. Really big jets. But success has taken a lot of the serendipity out of his life. Facebook is running him, as much as he's running Facebook.

While it can be really cool to be in the grip of some project you consider your life's work, there are advantages to serendipity. Among other things, it gives you more options to choose your life's work from. There's not even a trade off here you're not sacrificing anything if you forgo starting a start up at twenty because you will be more likely to succeed if you wait. In the unlikely case you are twenty in the astronomically case that you are twenty and you have some side project that takes off like Facebook did, then you face a choice to either be running with it or not and maybe it's reasonable to run with it. Usually the way that start ups take off is for the founders to make them take off. It's drastically stupid to do that at twenty.

Should you do it at any age? Starting a startup may sound kind of hard, if I haven't let me try again. Starting a startup is really hard. If it's too hard what if you are not up to this challenge?

The answer is the fifth counter intuitive point. You can tell. Your life so far has given you some idea of what your prospects might be if you wanted to become a mathematician or a professional football player. Boy, it's not every audience you can say that to. Unless you have had a very strange life indeed you have not done much that's like starting a startup. Meaning starting a startup will change you a lot if it works out. So what you're trying to estimate is not just what you are but what you can become and you can do that. Well, not me. for the last nine years it was my job to try to guess-I wrote predict in here and it came out as guess. Another Freudian slip. Seriously it's easy to tell how smart people are in
ten minutes. Hit a few tennis balls over the net, and do they hit them back at you or into the net? The hard part and the most important part was predicting how tough and ambitious they would become.

There may be no one at this point who has more experience than me in doing this. I can tell you how much an expert can know about that. The answer is not much. I learned from experience to keep completely open mind about which start up in each batch, which start ups would turn out to be the stars. The founders sometimes thought they knew. Some arrived feeling confident that they would ace Y Combinator just as they had aced every one of the few easy artificial tests they had faced in life so far. Others arrived wondering what mistake had caused them to be admitted and hoping that no one discover it.

There is little to no correlation between these attitudes and how things turn out. I've read the same is true in the military. The swaggering recruits are no more than likely to turn out to be really tough than the quiet ones and probably for the same reason. The tests are so different from tests in people's previous lives. If you are absolutely terrified of starting a startup you probably shouldn't do it. Unless you are one of those people who gets off on doing things you're afraid of. Otherwise if you are merely unsure of whether you are going to be able to do it, the only way to find out is to try, just not now.

So if you want to start a startup one day, what do you do now in college? There are only two things you need initially, an idea and cofounders. The MO for getting both of those is the same which leads to our sixth and last counterintuitive point.

The way to get start up ideas is not to try to think of startup ideas. I have written a whole essay on this and I am not going to repeat the whole thing here. But the short version is that if you make a conscious effort to try to think of startup ideas, you will think of ideas that are not only bad but bad and plausible sounding. Meaning you and everybody else will be fooled by them. You'll waste a lot of time before realizing they're no good. The way to come up with good startup ideas is to take a step back. Instead of trying to make a conscious effort to think of startup ideas, turn your brain into the type that has startup ideas unconsciously. In fact, so unconsciously that you don't even realize at first that they're startup ideas. This is not only possible. Yahoo. Google. Facebook. Apple all got started this way. None of these companies were supposed to be companies at first, they were all just side projects. The very best ideas almost always have to start as side projects because they're always such outliers that your conscious mind would reject them as ideas for companies.

How do you turn your mind into the kind that has startup ideas unconsciously? One, learn about a lot of things that matter. Two, work on problems that interest you. Three, with people you like and or respect. That's 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 is too narrow.

What was special about Brain Chesky and Joe Gebbia from Airbnb was not that they were experts in technology. They went to art school, they were experts in design. Perhaps more importantly they were really good at organizing people in getting projects done. So you don't have to work on technology per se, so long as you work on things that stretch you.

What kinds of things are those? Now that is very hard to answer in the general case. History is full of examples of young people who were working on problems that no one else at the time thought were important. In particular that their parents didn't think were important. On the other hand, history is even fuller of examples of parents that thought their kids were wasting their time and who were right.
How do you know if you’re working on real stuff? I mean when Twitch TV switched from being Justin.tv to Twitch TV and they were going to broadcast people playing video games. I was like what, but turned out to be a good business. I know how I know real problems are interesting and I am self-indulgent. I always like working on anything interesting things even if no one cares about them. I 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 things just because I was interested and they turned out to be useful later in some worldly way.

Y Combinator itself is something I only did because it seemed interesting. I seemed to have some internal compass that helps me out. This is for you not me and I don't know what you have in your heads. Maybe if I think more about it I can come up some juristics for recognizing genuine ally interesting ideas. For now all I can give you is the hopelessly question begging advice. Incidentally this is the actual meaning of the phrase begging the question. Hopelessly question begging advice that if you're interested in genially interesting problems.

Gratifying your interest energetically is the best way to prepare yourself for a start up and probably best way to live. Although I can't explain in the general case what counts as an interesting problem I can tell you about a large subs them (sp?). If you think of technology as something that’s spreading like some fractal sting, every point on the edge represents an interesting problem. Steam engine not so much maybe you never know. One guaranteed way to turn your mind into the type to start up ideas for them unconsciously. Is to get yourself to the leading edge of some technology. To as Baahooit (sp?) put it, "Live in the future." Uncannily appreciaent (sp?) to other people will seem obvious to you. You will may not realize they're start up ideas, but you will know they are 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. It wasn't meant to be a start up never tried to turn it into one. He just wanted to talk to his girlfriend in Taiwan without paying for long distance calls. Since He was an expert on networks. It seemed obvious to him that thing to do was to turn the sound into packets and ship them over the internet for free. Why didn't everybody do this? They were not good at writing this type of software. He never did anything with this. He never tried to turn this into a start up. That is how the best start ups tend to happen. Strangely enough the optimal thing to do in college

[00:30:11.18] Is not some sort of new vocational version of college focused on entrepreneurship its the classic version of college is education its own sake if you want to start your own startup[00:30:28.19] 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 inequations the component of entrepreneurship can never quite say that word with a straight face but really matters is domain expertise Larry page is Larry page be he was an expert on search and the way he became an expert of search [00:30:50.29] was because he was genuinely interested and not because of some ulterior motive[00:30:55.11] at its best starting a startup is merely a ulterior motive for cursorily and you’ll do it best if you introduce the ulterior motive at the end of the process so here is ultimate advice for young would be startup founders reduced to two words just learn. Alright how much time do we have left? Eighteen minutes for questions good god.

Do you guys have the questions?

Q: How can a nontechnical founder most efficacy contribute to a startup

A: If the startup is, if the startup is working in some domain if it’s not a pure technology startup. But it is working in some very specific domain if it is umber and in a non technical founder that was an expert in the limo business then actually then the non technical founder would be doing most of the work. Recruiting drivers and doing w/e else umber has to do and the technical founder would be just writing
the iphone app which probably less iphone and android which is less than half of it. If it's purely a technical start up the non technical founder does sales and brings coffee and cheeseburger to the programmer.

Q: Do you see any value in business school for people who pursue a entrepreneurship?

A: Basically know, it sound undiplomatic business school was designed for was to teach people management. Management is a problem that you only if you have in a start up is that if you are efficiently successful what you need to know early on to make a start up successful is developing products. You would be better off going to design school if you would want to go to some sort of school. Although frankly the way to learn how to do it is just to do it. One of the things I got wrong early on is that I advised people who were interested in starting a startup to go work for somebody other company for a few years before starting their own. Honestly the best way to learn on how to start a startup is just too just try to start it. You may not be successful but you will learn faster if you just do it. Business schools are trying really hard to do this. They were designed to train the officer core of large companies, which is what business seemed to be back when it was a choice to be either the officer core of large companies or Joe's Shoe Store. Then there was this new thing, apple that started as small as Joe's Shoe Store and turns into this giant mega company but they were not designed for that world they are good at what they're good at. They should just do that and screw this whole entrepreneurship thing.

Q: Management is a problem only if you are successful. What about those first two or three people?

A: Ideally you are successful before you even hire two or three people. Ideally you don't even have two or three people for quite awhile. When you do the first hires in a startup they are almost like founders they should be motivated by the same things, they can't be people you have to manage. This is not like the office, these have to be your peers, you shouldn't have to manage them much.

Q: So is it just a big no no, someone has to be managed no way they should be on the founding team.

A: In the case were you are doing something were you need some super advanced technical thing and there is some bowfin that knows this thing and no one else in this world including on how to wipe his mouth. It may be to your advantage to hire said Baffin and wipe his mouth for him. As a general rule you want people who are self motivated early on they should just be like founders.

Q: Do you think we are currently in a bubble?

A: I'll give you two answers to this question. One, ask me questions that are useful to this audience cause these people are here to learn how to start startups, and I have more data in my head than anybody else and your asking me questions a reporter does because they cannot think of anything interesting to ask. I will answer your question. There is a difference between prices merely being high and a bubble. A bubble is a very specific form of prices being high were people knowingly are pay high prices for something in the hope that they will be able to unload it later on some greater fool. That's what happened in the late 90's, in VC's knowly invested in bull shit startups thinking that they would be able to take those things public and unload the montreal investors before everything blew up and I was there I was there for that at the epicenter of it all. That is not what is happening today. Prices are high, valuations are high, valuations being high does not mean a bubble every commodity has prices that goes up and down in some sort of side wave defiantly prices are high. We tell people if you raise money don't think the next time you raise money it's going to be so easy, who knows maybe between now and then the Chinese economy will have exploded then there's a giant disaster recession assume the worst. Bubble, no.
Speaker/Q: I am seeing a trend among young people and successful entrepreneurship were they don’t want to start one great company but twenty and you are starting to see a rise in these labs attempts were they are going to try to launch a whole bunch of stuff, I don’t have any stellar examples yet.

Teacher: Do you mean ideal?

Speaker/Q: No, like ideal lab gear up camps new one, there’s nor flak.

Teach: There’s this new thing were people start labs that are supposed to spin off start ups. It might work, that’s how Twitter started in fact I meant Ideal Lab not idea that was another formalin slip. Twitter was not twitter at first twitter was the side project, side project at the company called Idea that was supposed to be in the podcasting business, and you like podcasting business, do those words even grammatically go together? The answer turned out to be no as Evan discovered. As the side project they spun off Twitter and boy was that a dog wagging tail, people are starting these things that are supposed to spin off start ups, will it work? Quite possibly if the right people do it. You can't do it though, because you have to do it with your own money.

Q: What advice do you have for female co-founders as they are pursuing funding?

A: Probably is true that women have a harder time raising money. I have noticed this empirically and Jessica is about to publish a bunch of interviews on female founders and a lot of them said that they thought they had a harder time raising money too. Remember I said the way to raise money? Make your start up actually do well that's just especially true in any case if you miss the ideal target from the VC's point of view in any respect. The way to solve that problem is make the startup do really well. In fact there was a point a year or two ago when I tweeted this growth graph of this company and I didn't say who they were. I knew it would get people to start asking and it was actually a female founded start up. That was having trouble raising money, but there growth graph was stupendous. So I tweeted it, knowing all these VC's would start asking me, “Who is that?” Growth graphs have no gender, so if they see the growth graph first, let them fall in love with that. Do well, which is generally good advice for all start ups.

Q: What would you learn in college right now?

A: Literary theory, no just kidding. Honestly, I think I might try and study physics that's the thing I feel I missed. For some reason, when I was a kid computers were the thing, maybe they still are. I got very excited learning to write code and you can write real programs in your bedroom. You can't build real accelerators, well maybe you can. Maybe physics I noticed I sort of look longley at physics so maybe. I don't know if that's going to be helpful starting a start up and I just told you to follow your own curiosity so who cares if it's helpful, it'll turn out to be helpful.

Q: What are your reoccurring systems in your work and personal life that make you efficient?

A: Having kids is a good way to be efficient. Because you have no time left so if you want to get anything done. The amount of none you do is high. Actually many parents, start up founders who have kids have made that point explicitly. They cause you to focus because you have no choice. I wouldn't actually recommend on having kids just to make you more focused. You know, I don't think I am very efficient, I have two ways of getting work done. One is during YCombinator, the way I worked on YCombinator's I was forced to. I had to set the application deadline, and then people would apply, and then there were all these applications that I had to respond to by a certain time. So I had to read them and I knew if I read them badly, we would get bad start ups so I tried really hard to read them well. So I set up this situation that forced me to work. The other kind of work I do is writing essays. And I do

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
that voluntarily, I am walking down the street and the essay starts writing itself in my head. I either force myself to work on less exciting things; I can't help working on exciting things. I don't have any useful techniques for making myself efficient. If you work on things you like, you don't have to force yourself to be efficient.

Q: When is a good time to turn a side project into a start up?

A: You will know, right. So the question is when you turn a side project into a start up, you will know that it is becoming a real start up when it takes over a alarming large percentage of your life, right. My god I've just spent all day working on this thing that's supposed to be a side project, I am going to fail all of my classes what am I going to do, right. Then maybe it's turning into a start up.

Speaker/Q: I know you talked a lot, earlier, about you'll know if your start up is doing extremely well, but I feel like a lot of pieces, is just a crayon [00:44:09.14]. What would you do or what would you recommend in those situations considering allocating time and resources, how do you balance?

A: When a start up is growing but not much. Didn't you tell them they were supposed to redo things that don't sell? You sir have not done the readings, you are busted. They're four, I wrote a whole essay answered that question and that is it do things that don't sell. Just go read that, because I can't remember everything I said. It's about exactly that problem.

Q: What kind of startup should not go through incubation, in your opinion?

A: Defiantly any that will fail. And or if you'll succeed but your intolerable person. That also sam sp? [00:45:07.25] would probably sooner do without. Short of that, I cannot think of any because a large percentage founder's are often surprised by how large a percentage of the problems that start ups have are the same regardless of what type of thing they're working on. And those tend to be kind of problems that YC helps the most not the ones that are domain specific. Can you think of the class of startups? That YC wouldn't work for? We had vision and fusion start ups in the last batch.

Q: You mentioned that it's good advice to learn a lot about something that matters, what are some good strategies to figure out what matters?

A: If you think of technology as something that's spreading as a sort of fractal sting. Anything on the edge represents an interesting idea, sounds familiar. Like I said that was the problem you have correctly identified the thing I didn't really answer the question were I gave this question begging answer sp? [00:46:16.00] I said I'm interested in interesting things and you said you were interested in interesting things, work on them and things will work out. How do you tell what is a real problem? I don't know that's like important enough to write a whole essay about. I don't know the answer and I probably should write something about that but I don't know. I figured out a technique to detecting whether you have a taste for genially interesting problems. Which is whether you find working on boring things intolerable and there are known boring things. Like literary theory and middle management in some large company. So if you can tolerate those things, then you must have stupendous self-discipline or you don't have a taste for genially interesting problems and vice versa.

Q: [00:47:08.03] Do you like SnapChat?

Q: If you hire people you like, you might get a monoculture and how do you deal with the blind spots that arise?

A: Starting a startup is where many things will be going wrong. You can't expect it to be perfect. The advantage is of hiring people you know and like are far greater than the small disadvantage of having some monoculture. You look at it empirically all the most successful start ups someone just hires all there pals out of college. Alright you guys thank you.
Lecture 3: How to Get Startup Ideas

http://www.paulgraham.com/startupideas.html

Paul Graham

The way to get startup ideas is not to try to think of startup ideas. It's to look for problems, preferably problems you have yourself.

The very best startup ideas tend to have three things in common: they're something the founders themselves want, that they themselves can build, and that few others realize are worth doing. Microsoft, Apple, Yahoo, Google, and Facebook all began this way.

Problems

Why is it so important to work on a problem you have? Among other things, it ensures the problem really exists. It sounds obvious to say you should only work on problems that exist. And yet by far the most common mistake startups make is to solve problems no one has.

I made it myself. In 1995 I started a company to put art galleries online. But galleries didn't want to be online. It's not how the art business works. So why did I spend 6 months working on this stupid idea? Because I didn't pay attention to users. I invented a model of the world that didn't correspond to reality, and worked from that. I didn't notice my model was wrong until I tried to convince users to pay for what we'd built. Even then I took embarrassingly long to catch on. I was attached to my model of the world, and I'd spent a lot of time on the software. They had to want it!

Why do so many founders build things no one wants? Because they begin by trying to think of startup ideas. That m.o. is doubly dangerous: it doesn't merely yield few good ideas; it yields bad ideas that sound plausible enough to fool you into working on them.

At YC we call these "made-up" or "sitcom" startup ideas. Imagine one of the characters on a TV show was starting a startup. The writers would have to invent something for it to do. But coming up with good startup ideas is hard. It's not something you can do for the asking. So (unless they got amazingly lucky) the writers would come up with an idea that sounded plausible, but was actually bad.

For example, a social network for pet owners. It doesn't sound obviously mistaken. Millions of people have pets. Often they care a lot about their pets and spend a lot of money on them. Surely many of these people would like a site where they could talk to other pet owners. Not all of them perhaps, but if just 2 or 3 percent were regular visitors, you could have millions of users. You could serve them targeted offers, and maybe charge for premium features. [1]

The danger of an idea like this is that when you run it by your friends with pets, they don't say "I would never use this." They say "Yeah, maybe I could see using something like that." Even when the startup launches, it will sound plausible to a lot of people. They don't want to use it themselves, at least not right now, but they could imagine other people wanting it. Sum that reaction across the entire population, and you have zero users.[2]

Well

When a startup launches, there have to be at least some users who really need what they're making—not just people who could see themselves using it one day, but who want it urgently. Usually this initial group of users is small, for the simple reason that if there were something that large numbers of people urgently needed and that could be built with the amount of effort a startup usually puts into a version one, it would probably already exist. Which means you have to compromise on one dimension: you can either build something a large number of people want a small amount, or something a small number of people want a large amount. Choose the latter. Not all ideas of that type are good startup ideas, but nearly all good startup ideas are of that type.

Imagine a graph whose x axis represents all the people who might want what you're making and whose y axis represents how much they want it. If you invert the scale on the y axis, you can envision companies as holes. Google is an immense crater: hundreds of millions of people use it, and they need it a lot. A startup just starting out can't expect to excavate that much volume. So you have two choices about the shape of hole you start with. You can either dig a hole that's broad but shallow, or one that's narrow and deep, like a well.

Made-up startup ideas are usually of the first type. Lots of people are mildly interested in a social network for pet owners.

Nearly all good startup ideas are of the second type. Microsoft was a well when they made Altair Basic. There were only a couple thousand Altair owners, but without this software they were programming in machine language. Thirty years later Facebook had the same shape. Their first site was exclusively for Harvard students, of which there are only a few thousand, but those few thousand users wanted it a lot.

When you have an idea for a startup, ask yourself: who wants this right now? Who wants this so much that they'll use it even when it's a crappy version one made by a two-person startup they've never heard of? If you can't answer that, the idea is probably bad. [3]

You don't need the narrowness of the well per se. It's depth you need; you get narrowness as a byproduct of optimizing for depth (and speed). But you almost always do get it. In practice the link between depth and narrowness is so strong.
that it's a good sign when you know that an idea will appeal strongly to a specific group or type of user.

But while demand shaped like a well is almost a necessary condition for a good startup idea, it's not a sufficient one. If Mark Zuckerberg had built something that could only ever have appealed to Harvard students, it would not have been a good startup idea. Facebook was a good idea because it started with a small market there was a fast path out of. Colleges are similar enough that if you build a facebook that works at Harvard, it will work at any college. So you spread rapidly through all the colleges. Once you have all the college students, you get everyone else simply by letting them in.

Similarly for Microsoft: Basic for the Altair; Basic for other machines; other languages besides Basic; operating systems; applications; IPO.

Self

How do you tell whether there's a path out of an idea? How do you tell whether something is the germ of a giant company, or just a niche product? Often you can't. The founders of Airbnb didn't realize at first how big a market they were tapping. Initially they had a much narrower idea. They were going to let hosts rent out space on their floors during conventions. They didn't foresee the expansion of this idea; it forced itself upon them gradually. All they knew at first is that they were onto something. That's probably as much as Bill Gates or Mark Zuckerberg knew at first.

Occasionally it's obvious from the beginning when there's a path out of the initial niche. And sometimes I can see a path that's not immediately obvious; that's one of our specialties at YC. But there are limits to how well this can be done, no matter how much experience you have. The most important thing to understand about paths out of the initial idea is the meta-fact that these are hard to see.

So if you can't predict whether there's a path out of an idea, how do you choose between ideas? The truth is disappointing but interesting: if you're the right sort of person, you have the right sort of hunches. If you're at the leading edge of a field that's changing fast, when you have a hunch that something is worth doing, you're more likely to be right.

In Zen and the Art of Motorcycle Maintenance, Robert Pirsig says: You want to know how to paint a perfect painting? It's easy. Make yourself perfect and then just paint naturally. I've wondered about that passage since I read it in high school. I'm not sure how useful his advice is for painting specifically, but it fits this situation well. Empirically, the way to have good startup ideas is to become the sort of person who has them.

Being at the leading edge of a field doesn't mean you have to be one of the people pushing it forward. You can also be at the leading edge as a user. It was not so much because he was a programmer that Facebook seemed a good idea to Mark Zuckerberg as because he used computers so much. If you'd asked most 40 year olds in 2004 whether they'd like to publish their lives semi-publicly on the Internet, they'd have been horrified at the idea. But Mark already lived online; to him it seemed natural.

Paul Buchheit says that people at the leading edge of a rapidly changing field "live in the future." Combine that with Pirsig and you get: Live in the future, then build what's missing. That describes the way many if not most of the biggest startups got started. Neither Apple nor Yahoo nor Google nor Facebook were even supposed to be companies at first. They grew out of things their founders built because there seemed a gap in the world.

If you look at the way successful founders have had their ideas, it's generally the result of some external stimulus hitting a prepared mind. Bill Gates and Paul Allen hear about the Altair and think "I bet we could write a Basic interpreter for it." Drew Houston realizes he's forgotten his USB stick and thinks "I really need to make my files live online." Lots of people heard about the Altair. Lots forgot USB sticks. The reason those stimuli caused those founders to start companies was that their experiences had prepared them to notice the opportunities they represented.

The verb you want to be using with respect to startup ideas is not "think up" but "notice." At YC we call ideas that grow naturally out of the founders' own experiences "organic" startup ideas. The most successful startups almost all begin this way.

That may not have been what you wanted to hear. You may have expected recipes for coming up with startup ideas, and instead I'm telling you that the key is to have a mind that's prepared in the right way. But disappointing though it may be, this is the truth. And it is a recipe of a sort, just one that in the worst case takes a year rather than a weekend.

If you're not at the leading edge of some rapidly changing field, you can get to one. For example, anyone reasonably smart can probably get to an edge of programming (e.g. building mobile apps) in a year. Since a successful startup will consume at least 3-5 years of your life, a year's preparation would be a reasonable investment. Especially if you're also looking for a cofounder. [4]

You don't have to learn programming to be at the leading edge of a domain that's changing fast. Other domains change fast. But while learning to hack is not necessary, it is for the foreseeable future sufficient. As Marc Andreessen put it, software is eating the world, and this trend has decades left to run.

Knowing how to hack also means that when you have ideas, you'll be able to implement them. That's not absolutely
necessary (Jeff Bezos couldn’t) but it’s an advantage. It’s a big advantage, when you’re considering an idea like putting a college facebook online, if instead of merely thinking "That’s an interesting idea," you can think instead "That’s an interesting idea. I’ll try building an initial version tonight." It’s even better when you’re both a programmer and the target user, because then the cycle of generating new versions and testing them on users can happen inside one head.

**Noticing**

Once you’re living in the future in some respect, the way to notice startup ideas is to look for things that seem to be missing. If you’re really at the leading edge of a rapidly changing field, there will be things that are obviously missing. What won’t be obvious is that they’re startup ideas. So if you want to find startup ideas, don’t merely turn on the filter "What’s missing?" Also turn off every other filter, particularly "Could this be a big company?" There’s plenty of time to apply that test later. But if you’re thinking about that initially, it may not only filter out lots of good ideas, but also cause you to focus on bad ones.

Most things that are missing will take some time to see. You almost have to trick yourself into seeing the ideas around you.

But you know the ideas are out there. This is not one of those problems where there might not be an answer. It’s impossibly unlikely that this is the exact moment when technological progress stops. You can be sure people are going to build things in the next few years that will make you think "What did I do before x?"

And when these problems get solved, they will probably seem flaimingly obvious in retrospect. What you need to do is turn off the filters that usually prevent you from seeing them. The most powerful is simply taking the current state of the world for granted. Even the most radically open-minded of us mostly do that. You couldn’t get from your bed to the front door if you stopped to question everything.

But if you’re looking for startup ideas you can sacrifice some of the efficiency of taking the status quo for granted and start to question things. Why is your inbox overflowing? Because you get a lot of email, or because it’s hard to get email out of your inbox? Why do you get so much email? What problems are people trying to solve by sending you email? Are there better ways to solve them? And why is it hard to get emails out of your inbox? Why do you keep emails around after you’ve read them? Is an inbox the optimal tool for that?

Pay particular attention to things that chafe you. The advantage of taking the status quo for granted is not just that it makes life (locally) more efficient, but also that it makes life more tolerable. If you knew about all the things we’ll get in the next 50 years but don’t have yet, you’d find present day life pretty constraining, just as someone from the present would if they were sent back 50 years in a time machine. When something annoys you, it could be because you’re living in the future.

When you find the right sort of problem, you should probably be able to describe it as obvious, at least to you. When we started Viaweb, all the online stores were built by hand, by web designers making individual HTML pages. It was obvious to us as programmers that these sites would have to be generated by software. [5]

Which means, strangely enough, that coming up with startup ideas is a question of seeing the obvious. That suggests how weird this process is: you’re trying to see things that are obvious, and yet that you hadn’t seen.

Since what you need to do here is loosen up your own mind, it may be best not to make too much of a direct frontal attack on the problem—i.e. to sit down and try to think of ideas. The best plan may be just to keep a background process running, looking for things that seem to be missing. Work on hard problems, driven mainly by curiosity, but have a second self watching over your shoulder, taking note of gaps and anomalies. [6]

Give yourself some time. You have a lot of control over the rate at which you turn yours into a prepared mind, but you have less control over the stimuli that spark ideas when they hit it. If Bill Gates and Paul Allen had constrained themselves to come up with a startup idea in one month, what if they’d chosen a month before the Altair appeared? They probably would have worked on a less promising idea. Drew Houston did work on a less promising idea before Dropbox: an SAT prep startup. But Dropbox was a much better idea, both in the absolute sense and also as a match for his skills. [7]

A good way to trick yourself into noticing ideas is to work on projects that seem like they’d be cool. If you do that, you’ll naturally tend to build things that are missing. It wouldn’t seem as interesting to build something that already existed.

Just as trying to think up startup ideas tends to produce bad ones, working on things that could be dismissed as "toys" often produces good ones. When something is described as a toy, that means it has everything an idea needs except being important. It’s cool; users love it; it just doesn’t matter. But if you’re living in the future and you build something cool that users love, it may matter more than outsiders think. Microcomputers seemed like toys when Apple and Microsoft started working on them. I’m old enough to remember that era; the usual term for people with their own microcomputers was "hobbyists." BackRub seemed like an inconsequential science project. The Facebook was just a way for undergrads to stalk one another.

At YC we’re excited when we meet startup working on things that we could imagine know-it-alls on forums dismissing as toys. To us that’s positive evidence an idea is good.
If you can afford to take a long view (and arguably you can't afford not to), you can turn "Live in the future and build what's missing" into something even better: Live in the future and build what seems interesting.

School

That's what I'd advise college students to do, rather than trying to learn about "entrepreneurship." "Entrepreneurship" is something you learn best by doing it. The examples of the most successful founders make that clear. What you should be spending your time on in college is ratcheting yourself into the future. College is an incomparable opportunity to do that. What a waste to sacrifice an opportunity to solve the hard part of starting a startup—becoming the sort of person who can have organic startup ideas—by spending time learning about the easy part. Especially since you won't even really learn about it, any more than you'd learn about sex in a class. All you'll learn is the words for things.

The clash of domains is a particularly fruitful source of ideas. If you know a lot about programming and you start learning about some other field, you'll probably see problems that software could solve. In fact, you're doubly likely to find good problems in another domain: (a) the inhabitants of that domain are not as likely as software people to have already solved their problems with software, and (b) since you come into the new domain totally ignorant, you don't even know what the status quo is to take for granted.

So if you're a CS major and you want to start a startup, instead of taking a class on entrepreneurship you're better off taking a class on, say, genetics. Or better still, go work for a biotech company. CS majors normally get summer jobs at computer hardware or software companies. But if you want to find startup ideas, you might do better to get a summer job in some unrelated field. [8]

Or don't take any extra classes, and just build things. It's no coincidence that Microsoft and Facebook both got started in January. At Harvard that is (or was) Reading Period, when students have no classes to attend because they're supposed to be studying for finals. [9]

But don't feel like you have to build things that will become startups. That's premature optimization. Just build things. Preferably with other students. It's not just the classes that make a university such a good place to crank oneself into the future. You're also surrounded by other people trying to do the same thing. If you work together with them on projects, you'll end up producing not just organic ideas, but organic ideas with organic founding teams—and that, empirically, is the best combination.

Beware of research. If an undergrad writes something all his friends start using, it's quite likely to represent a good startup idea. Whereas a PhD dissertation is extremely unlikely to. For some reason, the more a project has to count as research, the less likely it is to be something that could be turned into a startup. [10] I think the reason is that the subset of ideas that count as research is so narrow that it's unlikely that a project that satisfied that constraint would also satisfy the orthogonal constraint of solving users' problems. Whereas when students (or professors) build something as a side-project, they automatically gravitate toward solving users' problems—perhaps even with an additional energy that comes from being freed from the constraints of research.

Competition

Because a good idea should seem obvious, when you have one you'll tend to feel that you're late. Don't let that deter you. Worrying that you're late is one of the signs of a good idea. Ten minutes of searching the web will usually settle the question. Even if you find someone else working on the same thing, you're probably not too late. It's exceptionally rare for startups to be killed by competitors—so rare that you can almost discount the possibility. So unless you discover a competitor with the sort of lock-in that would prevent users from choosing you, don't discard the idea.

If you're uncertain, ask users. The question of whether you're too late is subsumed by the question of whether anyone urgently needs what you plan to make. If you have something that no competitor does and that some subset of users urgently need, you have a beachhead. [11]

The question then is whether that beachhead is big enough. Or more importantly, who's in it: if the beachhead consists of people doing something lots more people will be doing in the future, then it's probably big enough no matter how small it is. For example, if you're building something differentiated from competitors by the fact that it works on phones, but it only works on the newest phones, that's probably a big enough beachhead.

Err on the side of doing things where you'll face competitors. Inexperienced founders usually give competitors more credit than they deserve. Whether you succeed depends far more on you than on your competitors. So better a good idea with competitors than a bad one without.

You don't need to worry about entering a "crowded market" so long as you have a thesis about what everyone else in it is overlooking. In fact that's a very promising starting point. Google was that type of idea. Your thesis has to be more precise than "we're going to make an x that doesn't suck" though. You have to be able to phrase it in terms of something the incumbents are overlooking. Best of all is when you can say that they didn't have the courage of their convictions, and that your plan is what they'd have done if they'd followed through on their own insights. Google was that type of idea too. The search engines that preceded them shied away from the most radical implications of what
they were doing—particularly that the better a job they did, the faster users would leave.

A crowded market is actually a good sign, because it means both that there’s demand and that none of the existing solutions are good enough. A startup can’t hope to enter a market that’s obviously big and yet in which they have no competitors. So any startup that succeeds is either going to be entering a market with existing competitors, but armed with some secret weapon that will get them all the users (like Google), or entering a market that looks small but which will turn out to be big (like Microsoft). [12]

Filters

There are two more filters you’ll need to turn off if you want to notice startup ideas: the unsexy filter and the schlep filter.

Most programmers wish they could start a startup by just writing some brilliant code, pushing it to a server, and having users pay them lots of money. They’d prefer not to deal with tedious problems or get involved in messy ways with the real world. Which is a reasonable preference, because such things slow you down. But this preference is so widespread that the space of convenient startup ideas has been stripped pretty clean. If you let your mind wander a few blocks down the street to the messy, tedious ideas, you’ll find valuable ones just sitting there waiting to be implemented.

The schlep filter is so dangerous that I wrote a separate essay about the condition it induces, which I called schlep blindness. I gave Stripe as an example of a startup that benefited from turning off this filter, and a pretty striking example it is. Thousands of programmers were in a position to see this idea; thousands of programmers knew how painful it was to process payments before Stripe. But when they looked for startup ideas they didn’t see this one, because unconsciously they shrank from having to deal with payments. And dealing with payments is a schlep for Stripe, but not an intolerable one. In fact they might have had net less pain; because the fear of dealing with payments kept most people away from this idea, Stripe has had comparatively smooth sailing in other areas that are sometimes painful, like user acquisition. They didn't have to try very hard to make themselves heard by users, because users were desperately waiting for what they were building.

The unsexy filter is similar to the schlep filter, except it keeps you from working on problems you despise rather than ones you fear. We overcame this one to work on Viaweb. There were interesting things about the architecture of our software, but we weren’t interested in ecommerce per se. We could see the problem was one that needed to be solved though.

Turning off the schlep filter is more important than turning off the unsexy filter, because the schlep filter is more likely to be an illusion. And even to the degree it isn’t, it’s a worse form of self-indulgence. Starting a successful startup is going to be fairly laborious no matter what. Even if the product doesn't entail a lot of schleps, you'll still have plenty dealing with investors, hiring and firing people, and so on. So if there's some idea you think would be cool but you're kept away from by fear of the schleps involved, don't worry: any sufficiently good idea will have as many.

The unsexy filter, while still a source of error, is not as entirely useless as the schlep filter. If you’re at the leading edge of a field that's changing rapidly, your ideas about what's sexy will be somewhat correlated with what's valuable in practice. Particularly as you get older and more experienced. Plus if you find an idea sexy, you'll work on it more enthusiastically. [13]

Recipes

While the best way to discover startup ideas is to become the sort of person who has them and then build whatever interests you, sometimes you don't have that luxury. Sometimes you need an idea now. For example, if you’re working on a startup and your initial idea turns out to be bad.

For the rest of this essay I'll talk about tricks for coming up with startup ideas on demand. Although empirically you’re better off using the organic strategy, you could succeed this way. You just have to be more disciplined. When you use the organic method, you don't even notice an idea unless it's evidence that something is truly missing. But when you make a conscious effort to think of startup ideas, you have to replace this natural constraint with self-discipline. You'll see a lot more ideas, most of them bad, so you need to be able to filter them.

One of the biggest dangers of not using the organic method is the example of the organic method. Organic ideas feel like inspirations. There are a lot of stories about successful startups that began when the founders had what seemed a crazy idea but "just knew" it was promising. When you feel that about an idea you've had while trying to come up with startup ideas, you’re probably mistaken.

When searching for ideas, look in areas where you have some expertise. If you're a database expert, don't build a chat app for teenagers (unless you're also a teenager). Maybe it's a good idea, but you can't trust your judgment about that, so ignore it. There have to be other ideas that involve databases, and whose quality you can judge. Do you find it hard to come up with good ideas involving databases? That's because your expertise raises your standards. Your ideas about chat apps are just as bad, but you're giving yourself a Dunning-Kruger pass in that domain.

The place to start looking for ideas is things you need. There must be things you need. [14]

One good trick is to ask yourself whether in your previous job you ever found yourself saying "Why doesn't someone make x? If someone made x we'd buy it in a second." If you

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
can think of any x people said that about, you probably have
an idea. You know there's demand, and people don't say
that about things that are impossible to build.

More generally, try asking yourself whether there's
something unusual about you that makes your needs
different from most other people's. You're probably not the
only one. It's especially good if you're different in a way
people will increasingly be.

If you're changing ideas, one unusual thing about you is the
idea you'd previously been working on. Did you discover any
needs while working on it? Several well-known startups
began this way. Hotmail began as something its founders
wrote to talk about their previous startup idea while they
were working at their day jobs. [15]

A particularly promising way to be unusual is to be young.
Some of the most valuable new ideas take root first among
people in their teens and early twenties. And while young
founders are at a disadvantage in some respects, they're the
only ones who really understand their peers. It would have
been very hard for someone who wasn't a college student to
start Facebook. So if you're a young founder (under 23 say),
are there things you and your friends would like to do that
current technology won't let you?

The next best thing to an unmet need of your own is an
unmet need of someone else. Try talking to everyone you
can about the gaps they find in the world. What's missing?
What would they like to do that they can't? What's tedious or
annoying, particularly in their work? Let the conversation get
general; don't be trying too hard to find startup ideas. You're
just looking for something to spark a thought. Maybe you'll
notice a problem they didn't consciously realize they had,
because you know how to solve it.

When you find an unmet need that isn't your own, it may be
somewhat blurry at first. The person who needs something
may not know exactly what they need. In that case I often
recommend that founders act like consultants—that they do
what they'd do if they'd been retained to solve the problems
of this one user. People's problems are similar enough that
nearly all the code you write this way will be reusable, and
whatever isn't will be a small price to start out certain that
you've reached the bottom of the well. [16]

One way to ensure you do a good job solving other people's
problems is to make them your own. When Rajat Suri of E la
Carte decided to write software for restaurants, he got a job
as a waiter to learn how restaurants worked. That may seem
like taking things to extremes, but startups are extreme. We
love it when founders do such things.

In fact, one strategy I recommend to people who need a new
idea is not merely to turn off their schleip and unsexy filters,
but to seek out ideas that are unsexy or involve schleps.
Don't try to start Twitter. Those ideas are so rare that you
can't find them by looking for them. Make something unsexy
that people will pay you for.

A good trick for bypassing the schleip and to some extent the
unsexy filter is to ask what you wish someone else would
build, so that you could use it. What would you pay for right
now?

Since startups often garbage-collect broken companies and
industries, it can be a good trick to look for those that are
dying, or deserve to, and try to imagine what kind of
company would profit from their demise. For example,
journalism is in free fall at the moment. But there may still be
money to be made from something like journalism. What
sort of company might cause people in the future to say "this
replaced journalism" on some axis?

But imagine asking that in the future, not now. When one
company or industry replaces another, it usually comes in
from the side. So don't look for a replacement for x; look for
something that people will later say turned out to be a
replacement for x. And be imaginative about the axis along
which the replacement occurs. Traditional journalism, for
example, is a way for readers to get information and to kill
time, a way for writers to make money and to get attention,
and a vehicle for several different types of advertising. It
could be replaced on any of these axes (it has already
started to be on most).

When startups consume incumbents, they usually start by
serving some small but important market that the big players
ignore. It's particularly good if there's an admixture of disdain
in the big players' attitude, because that often misleads
them. For example, after Steve Wozniak built the computer
that became the Apple I, he felt obliged to give his then-
employer Hewlett-Packard the option to produce it.
Fortunately for him, they turned it down, and one of the
reasons they did was that it used a TV for a monitor, which
seemed intolerably déclassé to a high-end hardware
company like HP was at the time. [17]

Are there groups of scruffy but sophisticated users like the
early microcomputer "hobbyists" that are currently being
ignored by the big players? A startup with its sights set on
bigger things can often capture a small market easily by
expending an effort that wouldn't be justified by that market
alone.

Similarly, since the most successful startups generally ride
some wave bigger than themselves, it could be a good trick
to look for waves and ask how one could benefit from them.
The prices of gene sequencing and 3D printing are both
experiencing Moore's Law-like declines. What new things
will we be able to do in the new world we'll have in a few
years? What are we unconsciously ruling out as impossible
that will soon be possible?

Organic

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
But talking about looking explicitly for waves makes it clear that such recipes are plan B for getting startup ideas. Looking for waves is essentially a way to simulate the organic method. If you’re at the leading edge of some rapidly changing field, you don’t have to look for waves: you are the wave.

Finding startup ideas is a subtle business, and that’s why most people who try fail so miserably. It doesn’t work well simply to try to think of startup ideas. If you do that, you get bad ones that sound dangerously plausible. The best approach is more indirect: if you have the right sort of background, good startup ideas will seem obvious to you. But even then, not immediately. It takes time to come across situations where you notice something missing. And often these gaps won’t seem to be ideas for companies, just things that would be interesting to build. Which is why it’s good to have the time and the inclination to build things just because they’re interesting.

Live in the future and build what seems interesting. Strange as it sounds, that’s the real recipe.

Notes

[1] This form of bad idea has been around as long as the web. It was common in the 1990s, except then people who had it used to say they were going to create a portal for x instead of a social network for x. Structurally the idea is stone soup: you post a sign saying “this is the place for people interested in x,” and all those people show up and you make money from them. What lures founders into this sort of idea are statistics about the millions of people who might be interested in each type of x. What they forget is that any given person might have 20 affinities by this standard, and no one is going to visit 20 different communities regularly.

[2] I’m not saying, incidentally, that I know for sure a social network for pet owners is a bad idea. I know it’s a bad idea the way I know randomly generated DNA would not produce a viable organism. The set of plausible sounding startup ideas is many times larger than the set of good ones, and many of the good ones don’t even sound that plausible. So if all you know about a startup idea is that it sounds plausible, you have to assume it’s bad.

[3] More precisely, the users’ need has to give them sufficient activation energy to start using whatever you make, which can vary a lot. For example, the activation energy for enterprise software sold through traditional channels is very high, so you’d have to be a lot better to get users to switch. Whereas the activation energy required to switch to a new search engine is low. Which in turn is why search engines are so much better than enterprise software.

[4] This gets harder as you get older. While the space of ideas doesn’t have dangerous local maxima, the space of careers does. There are fairly high walls between most of the paths people take through life, and the older you get, the higher the walls become.

[5] It was also obvious to us that the web was going to be a big deal. Few non-programmers grasped that in 1995, but the programmers had seen what GUIs had done for desktop computers.

[6] Maybe it would work to have this second self keep a journal, and each night to make a brief entry listing the gaps and anomalies you’d noticed that day. Not startup ideas, just the raw gaps and anomalies.

[7] Sam Altman points out that taking time to come up with an idea is not merely a better strategy in an absolute sense, but also like an undervalued stock in that so few founders do it.

There’s comparatively little competition for the best ideas, because few founders are willing to put in the time required to notice them. Whereas there is a great deal of competition for mediocre ideas, because when people make up startup ideas, they tend to make up the same ones.

[8] For the computer hardware and software companies, summer jobs are the first phase of the recruiting funnel. But if you’re good you can skip the first phase. If you’re good you’ll have no trouble getting hired by these companies when you graduate, regardless of how you spent your summers.

[9] The empirical evidence suggests that if colleges want to help their students start startups, the best thing they can do is leave them alone in the right way.

[10] I’m speaking here of IT startups; in biotech things are different.

[11] This is an instance of a more general rule: focus on users, not competitors. The most important information about competitors is what you learn via users anyway.

[12] In practice most successful startups have elements of both. And you can describe each strategy in terms of the other by adjusting the boundaries of what you call the market. But it’s useful to consider these two ideas separately.

[13] I almost hesitate to raise that point though. Startups are businesses; the point of a business is to make money; and with that additional constraint, you can’t expect you’ll be able to spend all your time working on what interests you most.

[14] The need has to be a strong one. You can retroactively describe any made-up idea as something you need. But do you really need that recipe site or local event aggregator as much as Drew Houston needed Dropbox, or Brian Chesky and Joe Gebbia needed Airbnb?
Quite often at YC I find myself asking founders "Would you use this thing yourself, if you hadn't written it?" and you'd be surprised how often the answer is no.

[15] Paul Buchheit points out that trying to sell something bad can be a source of better ideas:

"The best technique I've found for dealing with YC companies that have bad ideas is to tell them to go sell the product ASAP (before wasting time building it). Not only do they learn that nobody wants what they are building, they very often come back with a real idea that they discovered in the process of trying to sell the bad idea."

[16] Here's a recipe that might produce the next Facebook, if you're college students. If you have a connection to one of the more powerful sororities at your school, approach the queen bees thereof and offer to be their personal IT consultants, building anything they could imagine needing in their social lives that didn't already exist. Anything that got built this way would be very promising, because such users are not just the most demanding but also the perfect point to spread from.

I have no idea whether this would work.

[17] And the reason it used a TV for a monitor is that Steve Wozniak started out by solving his own problems. He, like most of his peers, couldn't afford a monitor.

Thanks to Sam Altman, Mike Arrington, Paul Buchheit, John Collison, Patrick Collison, Garry Tan, and Harj Taggar for reading drafts of this, and Marc Andreessen, Joe Gebbia, Reid Hoffman, Shel Kaphan, Mike Moritz and Kevin Systrom for answering my questions about startup history.
always be opportunity for young companies, young people to innovate. As it should be.

Q: And that was going to be my closing question before I gave you chance to sort of free associate on your own. That is to talk to young people who sort of look to you as a role model. Opportunities for innovation you think they're still possible. What are the factors of success for young people today? What should they avoid?

A: I get asked this a lot and I have a pretty standard answer which is, a lot of people come to me and say "I want to be an entrepreneur." And I go "Oh that's great, what's your idea?" And they say "I don't have one yet." And I say "I think you should go get a job as a busboy or something until you find something you're really passionate about because it's a lot of work."

I'm convinced that about half of what separates the successful entrepreneurs from the non-successful ones is pure perseverance. It is so hard. You put so much of your life into this thing. There are such rough moments in time that I think most people give up. I don't blame them. It's really tough and it consumes your life. If you've got a family and you're in the early days of a company, I can't imagine how one could do it. I'm sure it's been done but it's rough. It's pretty much an eighteen hour day job, seven days a week for awhile. Unless you have a lot of passion about this, you're not going to survive. You're going to give it up. So you've got to have an idea, or a problem or a wrong that you want to right that you're passionate about, otherwise you're not going to have the perseverance to stick it through. I think that's half the battle right there.

Q: Your talking made me think of the other side of that. You talk about the passion side. What would you say, there's passion and then there's power. What you would say about the responsibilities of power, once you've achieved a certain level of success?

A: Power? What is that?

Q: You need passion to build a company like Apple or IBM or any other major company. Once you've taken the passion to that level and built a company and are in the position like a Bill Gates at Microsoft or anybody else, yourself, what are the responsibilities of those who have succeeded and have economic power, social power? I mean, you've changed the world. What are your responsibilities within that?

A: That question can be taken on many levels. Obviously if you're running a company you have responsibilities but as an individual I don't think you have responsibilities. I think the work speaks for itself. I don't think that people have special responsibilities just because they've done something that other people like or don't like. I think the work speaks for itself.

I think people could choose to do things if they want to but we're all going to be dead soon, that's my point of view. Somebody once told me, they said "Live each day as if it would be your last and one day you'll certainly be right." I do that. You never know when you're going to go but you are going to go pretty soon. If you're going to leave anything behind it's going to be your kids, a few friends and your work. So that's what I tend to worry about. I don't tend to think about responsibility. A matter of fact I tend to like to on occasion pretend I don't have any responsibilities. I try to remember the last day when I didn't have anything to do and didn't have anything to do the following day that I had to do and I had no responsibilities. It was decades ago. I pretend when I want to feel that way. I don't think in those terms. I think you have a responsibility to do really good stuff and get it out there for people to use and let them build on the shoulders of it and keep making better stuff.

Q: So the responsibility is to yourself and your own standards.

A: In our business, one person can't do anything anymore. You create a team of people around you. You have a responsibility of integrity of work to that team. Everybody does try to turn out the best work that they can.

Q: Any final comments or thoughts either for the record or off the record?

A: No. Not really. Timeframe's an interesting thing when you think about people looking back. I do think when people look back on this in a hundred years, they're going to see this as a remarkable time in history. And especially this area believe it or not. When you think of the innovation that's come out of this area, Silicon Valley and the whole San Francisco Berkeley Bay area, you've got the invention of the integrated circuit, the invention of the microprocessor, the invention of semiconductor memory, the invention of the modern hard disk drive, the invention of the modern floppy disk drive, the invention of the personal computer, invention of genetic engineering, the invention of object-oriented technology, the invention of graphical user interfaces at PARC, followed by Apple, the invention of networking. All that happened in this Bay area. It's incredible.

Q: Yeah, why do you think it happened? Why here?

A: Why here? Two or three reasons. I mean, this place is a remarkable place actually. You have to go back a little bit in history. This is where the beatnik era happened, San Francisco, which was a pretty interesting thing. This is where the hippie movement happened. This is the only place in America where Rock n Roll really happened. I mean all of the bands, most of the bands outside of Bob Dylan in the sixties, they all came outta here. From Joan Baez to Jefferson Airplane to the Grateful Dead, everything came outta here. Janis Joplin, Jimi Hendrix, everybody. Why is that? That's a little strange when you think about it.
Then you've also had Stanford and Berkeley, two awesome universities drawing smart people from all around the world and depositing them in this clean, sunny, nice place where there's a whole bunch of other nice people and pretty good food. At times, a lot of drugs and a lot of fun things to do. And so they stayed. And so there's a lot of human capital pouring in here. A lot of human capital. The average, you know, people seem pretty bright here relative to the rest of the country and people seem pretty open minded here relative to the rest of the county. And I think it's just a very unique place. And I think it's got the track record to prove it. So that tends to attract more people, too. And I give a lot of credit to the universities, probably the most credit to Stanford and Berkeley. And UC Cal.

Thanks.
4. Building product, talking and growing
Adora Cheung

Thanks for having me. Today I am going to be talking about how to go from zero users to many users. I'm just assuming that you have many great ideas in your head at this moment and you are thinking about what the next step is.

I wrote this up early this morning and a lot of this is based off of mistakes I have made in the past. So as Sam mentioned I went to YC in 2010 and I spent three years going back and forth, pivoting a bunch of times, starting over a bunch of times, and I have learned a lot about what not to do if I were to start another startup after Homejoy. A lot of it comes from failure and just telling you about what you shouldn't do, and kind of making generalizations of what you should do from that.

Just a reminder that you should take all advice as directionally good guidance, but every business is different. You're different, and I'm not you, so take everything with that in mind.

When you start a startup you should basically have a lot of time on your hands to concentrate on the startup. I'm not saying that you should quit school or quit work; what I'm saying is that you should have a lot of compressed time that is really dedicated to immersing yourself in the idea and developing solutions to the problem that you are trying to solve. So for example, if you're in school it is better to have one or two days straight per week to work on your idea versus spending two hours here and there every single day during the course of the week. Since this is an engineering class, it is sort of like coding. There is a lot of context switching and being able to really focus and immerse yourself is really important.

So, like I said, when I first wrote this up I was thinking what are the things that most people do incorrectly when starting a startup. The novice approach is sort of thinking, "I have this really great idea, I don't want to tell anyone about it. I'm going to build, build, build and then going to maybe tell one or two people and then I'm going to launch it on TechCrunch or somewhere like that, and then I'm going to get lots of users."

But what really happens is because you did not get a lot of the feedback, maybe you get a lot of people to your site, but no one sticks around because you didn't get that initial user feedback. If you're lucky enough to have some money in the bank you might go buy some users but it just whittles out over time and you just give up. It is sort of a vicious cycle. I actually did this once, and I did this while I was in YC. When I went through YC I didn't even launch a product. I didn't launch on TechCrunch which is something you should definitely do. You don't ever want to get into that cycle because you'll just end up with nothing good.

The next thing is that you have an idea and you should really think about what the idea is really solving. Like what is your actual problem. You should be able to describe your problem in one sentence. And then you should think, "How does that problem relate to me? And I'm really passionate about that problem?" And then you should think, "Okay it's a problem I have, but is it a problem that other people have?" And you verify that the going out and talking to people.
One of the biggest mistakes I've made involves my Co-founder and I, who is also my brother. We started a company called Pathjoy in 2009 or 2010. We had two goals in mind. One was to create a company that made people really happy, and to create a company that was very, very impactful. A good proxy for that is to just create a big huge company. And so we thought, okay, the problem we are solving is to make people happier. We first went to the notion of who are the people who make people happy. We came up with life coaches and therapist. It seemed kind of obvious to create a platform for life coaches and therapist. What happened as a result was that when we started using the product ourselves, we aren't cynical people by any means, but life coaches and therapists are just not people we would use ourselves. It was sort of useless to us. So it wasn't even a problem that we had and it wasn't something that we were super passionate about building out, yet we spent almost a year trying to do this. And so it you just start from T =0 and think about this before you build any product I think you can save yourself a lot of headache down the road from doing something you don't want to do.

So you have a problem and you are able to state it, where do you start and how do you think of solutions? The first thing you should do is think about the industry that you are getting yourself into. Whether it is big or whether it is huge should really immerse yourself into that industry. And there are a number of ways to do this. One is to really become a cog in that industry for a little bit. And so it might seem a little counterintuitive to do this because most people say that if you really want to disrupt an industry you should really not be a player in it. Someone who spent 20 or 30 years in an industry is probably set in their ways and just used to the way things work and really can't think about what the inefficiencies are or the things that you can "disrupt". However, as a newbie coming into the industry you really should take one or two months to just really understand what all of the little bits and pieces of the industry are and how it works. Because it's when you get into the details, that's when you start seeing things that you can be exploiting and things that are really inefficient and may provide a huge overhead cost that you may be able to cut down.

So an example of this is that when we started Homejoy we started with the cleaning industry, and when we started we were the cleaners ourselves. We started to clean houses and we found out really quickly that we were very bad cleaners. As a result, we said okay, we have to learn more about this and we went to buy books. We bought books about how to clean, which helped maybe a little bit. We learned a little more about cleaning supplies but it is sort of like basketball, you can read and learn about basketball but you're not going to get better at it if you don't actually train and throw a basketball into the.

And so we decided that one of us was going to have to learn how to clean. Or at least get trained by a professional. We actually went to get a job at a cleaning company itself. The cool thing was I learned how to clean from training the few weeks that I was there at the cleaning company, but the even better thing was that I learned a lot about how a local cleaning company works. In that sense I learned why a local cleaning company could not become huge like Homejoy is today. And that is because they are pretty old school and they have a lot of things that are done inefficiently. Such as booking the customer and optimizing the cleaners schedules.

If you are in a situation like mine where there is a service element of it then you should go and do that service yourself. If your thing is related to restaurants you should become a waiter, if it is related to painting become a painter and kind of get in the shoes of your customers from all angles of what you are trying to build.

The other thing is there is also a level of possessiveness that you should have with it as well. You should be so obsessed that you want to know what everybody in that space is doing. And it is things like writing a list of all of the potential competitors, similar types of companies and Google searching them and clicking on every single link and reading every single article from search result number 1 to 1000. I found all potential competitors big and small and if they were public I would go and read their
S-1s, I would go read all of their quarterly financials, I would sit on earnings calls - you know most of these you don't get much out of it but there are these golden nuggets that you will find every once in a while. And you won't be able to find that unless you actually go through the work of getting all that information in your head. You should become an expert in your industry. There should be no doubt when you are building this that you are the expert so that people trust you when you are building this product.

The second thing is identifying customer segments. Ideally at the end of the day you have built a product or business that everybody in the world is using. But realistically in the beginning you kind of want to corner off a certain part of the customer base so that you can really optimize for them. It is just about focus and whether you are catering to teenage girls or whether it is soccer moms you will be able to focus a lot on their needs.

And lastly, before you even create a product or before you put code down, you should really storyboard out the user experience of how you are going to solve the problem. And that is not just meaning the website itself, it also means how does the customer find out about you. It can be through an ad or word-of-mouth, and then they come to your site and they learn more about you. What does that text say and what are you communicating to them when they sign up for the project and when they purchase the service? What are they actually getting from your service or product? After they finish using the product or service do they leave a review or do they leave comments? You need to be able to go through that whole flow and visualize your head what the perfect user experience is. And then put it down on paper and put it into code, and then start from there.

So, you have all these ideas in your head now you kind of know what the core customer base is that you want to go after and you know everything about the industry, what do you do next? You start building your product. The common phrase that most people use today is, "You should build a minimum viable product." And I underlined viable because I think a lot of people skip that part and they go out with a feature and the whole user experience in the very beginning is flat. Minimal viable product pretty much means what is the smallest feature set that you should build to solve the problem that you are trying to solve. I think if you go through the whole storyboarding experience you can kind of figured that out very quickly. But again, you have to be talking to users you have to be seeing what exists out there already, and what you should be building to solve their immediate needs.

And the second thing is that before you put things in front of the user you should really have your product positioning down. What I mean by that is that you should be able to go to a person and be able to say, "Hey, this does XY and Z in one sentence." So for example at Homejoy we started off with something super complicated. We were an online platform for home services, we started with cleaning and you can choose blah blah blah. It just went on for paragraphs and paragraphs.

When we went to potential users to come on our platform they would kind of get bored after the first few sentences. What we found out was that we needed a one-liner in the one-liner was very important. It kind of describes the functional benefits of what you do. In the future when you are trying to build a brand or whatnot you should be able to describe what are the emotional benefits and stuff like that. But when you are starting with no users you really need to tell them what they are going to get out of it. After we changed our position to get your place clean for $20 an hour then everyone got it and we were able to get users in the door that way.

So you have a MVP going out there now how do you get your first few users to start trying it? The first few users should be obviously people you are connected with. You and your cofounder should be using it, your mom and dad should be using it, and your friends and coworkers should be using it. Beyond that you want to get more user feedback. I've listed here some of the obvious places to go to depending on what you are selling you can take your pick of the draw here. So, online communities, on Hacker

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
News now there is the show HN - that's a great place. Especially if you are building tools for developers and things like that. Local communities - so if you're building consumer products you know there are a lot of influential local community mailing lists. Especially those for parents. Those are places you might want to hit up too.

At Homejoy we actually tried all of these. We used it ourselves and that was fine. We were only cleaners so that was pretty easy. Our parents live in Milwaukee and we were based in Mountain View so that didn't work. Friends and coworkers were kind of like in San Francisco and elsewhere so we didn't have too many of them use it. So we actually ended up in a dead end of not being able to convince many people to use it in the beginning. So what we did was, because we are in Mountain View, some of you guys might know on Caster street they have street fairs there during the summertime. So we would go out and basically chased down people and get them to try to book a cleaning. Almost everyone would say no until one day we just took advantage. It was a very hot and humid day and what we noticed was that everyone gravitated towards the food and drink area, especially on a hot day. We figured we needed to get in the middle of that so we took water bottles and froze them and we started handing out free bottles of water that were cold. And people just came to us. I think we basically guilt tripped people into booking cleanings. But the proof in the pudding was that I figured most of the people were guilt tripped into doing it but when they went home and they didn't cancel on us. Well, some of them did but the majority of them did not. I thought that's good, I have to go clean their houses but at least there is something we are actually solving here.

I know another startup in the last batch, I forgot their name right now, but they were selling shipping type products or trying to replace shipping products. So they would show up to the US postal office and find people who were trying to ship products and just take them out of line and get them to try to use the product and have them ship it for them. So you just have to go to places where people are really going to show up. Your conversion rate is going to be really low but to go from 0 to 1 to 3 to 4 these are the kind of things you might have to do.

So now that you have users using you what do you do with all of these users? The first thing you should do is make sure that there is a way for people to contact you. Ideally there is a phone number and if you put up a phone number one good idea is to make sure that you have a voicemail so that you won't be picking it up all the time. But in any case a way for people to get inbound feedback is good but really what you should be doing is going out to your users and talking to them. Get away from your desk and just get out and do the work. It seems like a slog and it is going to be a slog but this is where you are going to get the best feedback ever for your product. And this is where it is going to teach you what features you need to completely change, get rid of, or what features you need to build.

So another thing that you should be tracking is how are you doing in general from a macro perspective. The best way to do that is by tracking customer retention. The number of people that came in the door today, the number of people who are coming back tomorrow, the next day and so forth. Usually over
time you are kind of looking at monthly retention so people who came in the door today are they still using it next month and so forth. The problem with that metric is that it takes forever to collect that data and sometimes you don't have a month or two months or three months to figure that out. So a good leading indicator is actually collecting reviews and ratings. Such as five-star and four-star reviews or collecting some notion of NPS, which is Net Promoter Score. So you're basically asking them for a rating from 0 to 10 about how likely are they to recommend you to a friend and calculating the NPS. Over time what you'll see is that as you are building new features you will be able to see that the reviews and the retention are going up over time. That means that you are doing a good job. If it is going down then you are doing a bad job. If it is kind of staying the same that probably means that you need to go out and figure out what new things you should be building.

One thing you should be wary of is the honesty curve. Some people will just lie to you. These are degrees of separation from you, and this is the level of honesty. So here this is your mom, these are the friend's of your friends and here are random people. Your mom will use your product and she will be proud of you anyway, so she'll be honest this much. Your friends will be pretty honest with you and give you feedback because they care about you - this is assuming this is a free product - and then over time as you get more and more random, these people don't know who you are. There are people over here who don't care about giving you feedback. So take this into consideration when getting user feedback. So say now this is a paid product. So when it is a paid product your mom is gone here. She is just going to lie to you and tell you it's great. But then it kind of goes like this (draws graph going upward). Your friends are going to support you and give you the right feedback but it is actually these random people out here that if they really don't think that what they paid for was worth it they care going to really tell you. That's because it is money out the door.

So this is another way of saying that you are going to get the best feedback if you just make someone pay for it. That's not to say that you should make people pay for it the first time out, but it is to say that if you are going to build a product that you are going to eventually need to pay for the software or for the hardware or whatever then get to the point where you can do that very fast. Because that is when you can get to the more meaty stuff of how you can get more paying users in the door.

You're getting a lot of feedback and what you do before you officially launch the product? You always want to be building fast and you want to be optimizing for this stage of your growth. You might have 10 users at this point and there is no point in trying to build features when you might have 10 million users. You want to optimize for the next stage of growth which will be 10 to 100 users. What are the features you really need that and just go with that. One of the things I found when building a marketplace is that process is very important over time as you scale.

You need not to try and automate everything and create software to have robots run everything. What you should do to really understand what you should build it manually do it yourself. An example of this is when we started taking on cleaning professionals on to our platform would ask them a bunch of questions over the phone and then in person would ask a bunch of questions as well. And then they would go to a test clean and then they would get onboarded to our platform if they were good enough. Doing all these questions for that many candidates we had a 3-5% acceptance rate. What happened over time was that we learned certain questions that we were asking were good indicators as to whether or not they would be a good or bad performer on the platform.

Through data collection and just looking at everything we could ask on an online form. That is when we put an online application, they could apply and then we would ask them maybe several other questions during the in person interview. If you try to automate things too fast then you run into this potential problem of not being able to move quickly and iterate things like questions on an application and things like that.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
A third point here is temporary brokenness is much better than permanent paralysis. By that what I mean is perfection is irrelevant during this stage. When you get to the next stage of growth what you are trying to perfect in one stage is not going to matter anyway. So do not worry about all of the edge cases when you are building something just worry about the generic case of who your core user is going to be. As you get bigger and bigger the volume of those edge cases increase over time and you will want to build for that.

Lastly beware of the Frankenstein approach which is - great you talked to all of these users and they gave you all of these ideas and the first thing you are going to want to do is go build every single one of them and then go show them the next day and make them happier. You should definitely listen to user feedback but when someone tells you to build a feature you shouldn't go build it right away. What you should really do is get to the bottom of why they are asking you to build the feature. Usually what they are suggesting is not the best idea. What they are really suggesting is that I have this other problem that you either created for me while using the product or I really need this problem solved if I'm going to pay to use this product. So figure that out first before piling on a bunch of features which then hides the problem altogether.

So you have a product that you are ready to ship - some people at this point will continue building the product and not ship it at all. I think the whole idea of being stealth and perfecting the product to no end is the idea that imitation is cheaper than innovation in terms of time and money and capital. I think that everyone should always assume in general that if you have a really good idea no matter when you launch someone is going to fast follow you and someone is going to execute as hard as they possibly can to catch up with you. There is no point in holding out on all of that user feedback that you can get by getting a lot of users because he felt paranoid that someone is going to do this to you.

I hate to keep harping on it but these are things that I see today with founders and something that I went through as well. And I think that unless you are building something that requires tens of millions of dollars just to start up there is really no point in waiting around to launch the product.

So say you have something that you feel ready to get lots of users on. So what do you do at this point?

I will go over various types of growth in the next slides, but the one thing to note here early on when it is just you, your cofounder, and a couple of other people building you aren't creating a team just for growth. It is going to be one person and one person only. You really need to focus and you are going to be tempted to try five different strategies at one time. But really what you should do is take one channel and really execute on it for an entire week and just focus on that. And if that works continue executing on it until it caps out. If it doesn't work then just move on. By doing this you will feel more certain that the channel you were working on is wrong and your initial hypothesis is wrong than if you only spent a third of your time on it over the course of a few weeks. So learn one channel at a time.

Second, when you find one channel at a time and strategies that work always be iterating on it. You can potentially create a playbook and give it to someone else to iterate on it but these channels always change. Anything from Facebook ads to Google ads, the environments that you don't control change all of the time and you should always be iterating and optimizing for that. In the beginning when you see a channel that [*31:22*] just to get rid of it and go on there are lots of other things to try. But over time go back to those channels and look at it again.

An example is that in the beginning at Homejoy we had no money so when we tried to buy Google ads to get users in the door quickly - what we found was that all of these national companies had more money than us, they were making a lot more money on the job than us. So they were able to acquire

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208) 73 / 412
users at a much higher cost than us. So we couldn't afford that and we had to go through another channel. But today we make more money on the job, and we are better at some things. So we should probably revisit the idea of buying Google ads. That's what I mean by that.

And the key to all of this is creativity. Performance marketing, or marketing and growth in general can be very technical but it is actually technical and you have to be creative because if it was really easy and bland then everyone would be growing right now. So you always have to find that little thing that no one else is doing and do that to the extreme.

So there are three types of growth. Sticky, viral, and paid growth. Sticky growth is trying to get your existing users to come back and pay you more or use you more. Viral growth is when people talk about you. So you use a product, you really like it and you tell ten other friends, and they like it. That's viral growth. And the third is paid growth. If you happen to have money in the bank you're going to be able to use part of that money to buy growth.

The central theme that I'm going to go through is sustainability. By sustainable growth I mean you are basically not a leaky bucket. The money you put in has a good return investment on it. So sticky growth is, like I said, trying to get your existing users to come back and buy stuff. The only thing that really matters here is that you deliver a good experience. Right? If you deliver a good experience people are going to want to keep using you. If you deliver an addictive experience people are going to want to keep using you. And the way to measure this and to really look at this and how you are doing over time with whether you are providing good sticky growth is to look at the clv and retention cohort analysis.

CLV, some people call it TLV is a customer's lifetime. It is basically the net revenue that a customer brings in the door over a period of time. So a 12 month clv is how much net revenue does a customer give you over 12 months. And sometimes people will do the months and six months and so forth. So when I say cohort basically what you are looking at is time, and this is percent of the users coming back to you. So at period zero you are at 100%.

So cohort is another name for customer segments. For example you might look at the female versus male cohorts or people in Atlanta, Georgia versus people in Sacramento, California cohorts. The most common one is by month. So cohort equals month and let's just say for this exercise we are looking at March 2012. So in March 2012, 100% of the people are using your product. Now, one month later 50% of the people might come back. Now, in the second month how many people that came in March are coming back two months later? That might be down. So over time you will have a curve that looks like this. There is always some initial drop off. The reasons that people don't stay after first use could be that it wasn't worth it or they had a bad experience, or something like that. And then over time what you want is for your curve to flatten out. These over here become your core customers. These are the ones that will stay with you over time.

Say we are at one year later and you have built a bunch of stuff. You grab out the same thing and hopefully what you see is that you have a curve like this. That is, that even in the first period more than 50% of the people came back to you and more and more people are sticking with you. A really bad retention curve looks like this - which is after the first use they just see you so much that no one even comes back. I don't know what kind of business that is, it is obviously a shitty business. I can't explain a good business that has a retention curve like that. Over time as you are thinking of strategies to increase this curve and to keep making it go up and up and up you want to keep looking at this analysis over time to see if that strategy is working for you.

The second kind of growth is viral growth. Like sticky growth you also need to deliver a good experience. But on top of that you need to deliver a really, really good experience. What is going to
make these people shout out loud on twitter or on Facebook or whatever and tell all their friends and email all of their family about you. You have to really deliver a good experience. Combined with that is you have to have really good mechanics for the referral program itself. You have 100 customers who really want to talk about you. Now how are they going to talk about you?

So in that sense the viral growth strategy is all about building a good experience, but if you have that how do you build a good referral program. I have listed the three main parts of that. One is the customer touch points which is where our people learning that they can refer other people? That might be after they book or after they sign up. A better one is after they use the product after a while and you see that they are highly engaged, then you show them that link and get them to send it out to everyone. Another one is if you are doing more of a platform type play - for Homejoy we actually go inside their home. So another customer touch point is when the cleaning professional is inside the home they can have a leave behind and we can show them something there too as well. You want to basically put the customer touch points and the actual link to however they are going to refer their friends at a point in time where they are highly engaged and you know that they are loving you.

The second is program mechanics. The most common thing I have seen is $10 for $10. You get $10 if you invite your friends and they use it and they get $10. And so you should try different types of mechanics in that sense and try to optimize for whatever works for you. It could be 25 for 25 or it could be 10 for 0, it could be any of these things. And lastly, when your friend clicks on your referral link, when they come back to the site it is really important to optimize that conversion flow of how they are going to sign up. Sometimes you need to sell them in a different manner or up-sell that a friend suggested that you use this and so forth. So with all of these combined you will really need to play around with them in different dimensions and come up with a good referral program.

And lastly it is paid growth. Some examples of paid growth is this right here. And these are some of the most obvious ones and I'm sure that you guys can think of more. Pain growth is you happen to have money you can spend - you may have credit cards or whatever - but you can spend something to get users. So the correct way to think about paid growth is that you are going to risk putting money out there so what are you going to get a return. The simple way to think about it - is your clv, your customer's lifetime - is it more than your cac. And your cac is an abbreviation for customer acquisition costs. So an example is - say you run a bunch of ads over 12 months and the customer is worth $300 to you. Each one of these ads, when you click on it the cpc costs different types of money, and then when they click on your ad they have to come to the site and sign up or buy something. And the conversion rates are different for all of these ads. The cac is calculated by the cpc divided by the conversion. So you see that there are different acquisition costs for different types of ads. To determine whether or not that is a good or bad ad all you have to do is clv minus the cac. If it is more than zero you are earning a profit. So you see that despite the clv remaining the same and the conversions being higher or lower sometimes some ads that might seem good actually don't seem so good at the end of the day. You can look at this for your whole entire customer base aggregating all of your customers together, but the better way of looking at it is, to break it down by customer segments. If you are building a marketplace for country music the clvs of someone in Nashville, Tennessee is going to be much larger than the clvs of someone in Czechoslovakia. I just assume that is the case anyway.

You will want to make sure that when you are buying ads for these different types of cohorts that you know what the differences are and you don't want to mix everything together. The last point on payback and sustainability - I think a lot of businesses get in trouble and they turn into bad businesses when they start spending beyond their means. And it has a lot to do with risk tolerance or how much risk you are willing to take on. So when you look at these clvs which is suppose you get a customer that is worth $300 after 12 months. In the first month they are worth $100. If you wait until the 12 month period then they give you the other $200. But if in the first period you are actually paying $200 for
them than you are in the hole for $100 until the end of the 12 month period. That's when you start to get into potentially unsustainable growth. Something could happen at the end of the 12 months where you don't actually get the $200 from the customer and you end up in a very bad situation. Essentially, at the end of the day you could be running out of money. And if you are doing this with credit cards you will definitely find that you are going to have to declare bankruptcy very soon.

So again, payback time is very important. Safe time to go with is three months. If you are very risk loving then maybe 12 months is better. Beyond 12 months is very much unsafe territory.

The art of pivoting - Homejoy in its current concept was literally the 13th idea we fully built out and tried to execute on and tried to get customers for. And so a lot of the questions I get are," How do you even get to that 13 idea, and how did you decide when to move on ?" The best guidance that I can give on that is the kind of look at these three criteria, which is once you realize that you can't grow, and despite building out all of these great features and talking to all of these users none of them stick, or the economics of the business just don't make sense - then once you make that realization you just need to move on.

I think the trickiest one is probably the growth one because there are so many stories out there where the founders stuck with the idea and then after three years all of a sudden it started growing. So the trick here is what you really should do is have a growth plan when you start out. What is an optimistic but realistic way to grow this business? it might look something like this. In week one you just want one user, in week two you want maybe two users and so forth. And you can keep doubling up and up. In week one you should do basically build as much as possible to get that one user. And then a week to build as much to get two users. If you have a product that people want you should be able to maintain this growth curve pretty easily by just walking around and manually finding people. It is when you get to 100 users a week when you need these growth strategies to start working. What I tell people is usually if you are fully executing on your product, and you are working really hard, then if you go three or four weeks in a row of no growth or backwards growth, then it is time to maybe consider a pivot. Maybe not in the sense that you completely come up with a new idea but you are probably fundamentally doing something wrong because of that early stage of a startup should always be growing. This is optimistically what it looks like and this is the kind of growth curve that I set forth and put out when I started Homejoy, but really what it looks like is like this. In week one you just want one user, in week two you want maybe two users and so forth. And you can keep doubling up and up.

I can take questions at this time.

Q: So one question online was if your users already have a product that they are already comfortable with how do you get them to switch to yours?

A: There is always a switchover cost. I will tell you the example of Homejoy. We were actually creating a new market in the sense that most of our users had never had cleanings before so it was pretty simple to get them on board. And a lot of people who have cleaners already really trust their cleaner. To get into common use something else is probably the most difficult task in the world. When you are building things and trying to get people to switch over to you what you really need to do is find the moments where your product or what you are offering is much better or very much differentiated from the existing solution they have. So an example is someone who had a regular cleaner and maybe had a party one day and they needed a cleaning almost the next day. Because Homejoy in most areas has next-day availability they would just come to Homejoy and use it because they knew they couldn't get their regular cleaner. And once they start using the product then that is when they start realizing the little advantages of using Homejoy which adds up to a big advantage. Leaving cash out or using checks was
really annoying so being able to do all of my payments was more convenient to me. Being able to cancel or reschedule according to your own schedule was very convenient. A lot of people when they build a product they are like - and these 50 things are better than the existing solution - and even if the benefits outweigh the switching costs it is really hard to actually tell that to a user and try to get them to aggregate all of those benefits over many little things. It is better to have one or two things that clearly differentiates your self from the other product done.
Cleaning sucks.

Luckily, I live in San Francisco where you can get anything as a service (e.g. lunch, dinner, whiskey, washing, your fridge… even WordPress support!). When I read about a service that offered pre-screened cleaners for $20 per hour, I tried it out the following week.

First I will share what I learned from founder and CEO Adeora Chung at Startup Grind, then I’ll give you my first-hand review of the Homejoy service.

Look for inefficiency

Adora’s brother and co-founder, Aaron, sounds like most 20-something guys I know. He lacks motivation in a very specific area of his life. Cleaning. He lived like a slob with Adora as his roommate, but finally caved and decided to hire a cleaner.

He quickly realized:
- Established cleaning companies were prohibitively expensive
- Cleaners from online classifieds were unscreened… and potentially axe-murderers

Adora realized that other people faced this problem. After some further research, Adora noted that most cleaning companies operate on tiny margins and pay their cleaners minimum wage. This is not because it’s a bad business, but rather, most cleaning companies are inefficient, relying on man-hours, instead of systems. Adora dug deeper, finding more and more broken business processes. From candidate screening to recruitment to training, the smaller cleaning companies were operationally challenged. Like all good engineers, Adora saw this as a system with inputs and outputs.

Adora had finally found a problem that could be solved with a better system.

Keep it simple

When you’re starting out with a new business, it’s difficult to be concise, clear and specific about who you are, what you do and what problem you solve. Some founders rely on the we’re X for Y formula, kind of like… “Spotify meets Grinder, but for rental cars.”

Other founders will give you a verbose explanation, which is well done by one of our favourite HBO shows:

**Pied Piper** is a multi-platform technology based on a proprietary universal compression algorithm that has consistently fielded high Weisman Scores™ that are not merely competitive, but approach the theoretical limit of lossless compression.

Adora and her brother stumbled and faltered on their offer multiple times. At one point, Adora rewrote gossip content on a media site to generate a small amount of revenue to keep their lights on, while they hacked away on project after project. Adora and Aaron toyed with the idea of an online video coaching support service, but serendipitously landed on their house cleaning platform idea.

An early customer research game changer was asking potential customers a very simple question.

**Would you pay $20 per hour for a reliable cleaner?**

The overwhelming response was YES! They had a clear offer, a captive market and a big idea of how to do it. Now, they needed to execute. They set out on their path to deliver a house clean from just $20 per hour.

Growth hack at the grassroots

HomeJoy acquired their early customers the old fashioned way. Instead of trying to game SEO or launch an advertising campaign, they attended local fairs in their Palo Alto neighborhood and distributed flyers about their service.

Was it scalable? No. Was it successful? Not the first time around.

Their flyers tanked, but they came up with a simple idea that grabbed people’s attention.
Palo Alto gets hot. Fair patrons get thirsty. Problem = solution. Adora and Aaron handed out free bottles of icy cold water with Homejoy flyers.

This tiny tweak helped to attract their first customers. Simple ideas like this can generate enough interest to get your business off the ground.

Learn the ropes

Great entrepreneurs are self aware.

Adora knew that she wasn’t a natural born cleaner, but realized she needed to learn how to clean to run a cleaning company. Instead of avoiding the discomfort of scrubbing toilets and cleaning ovens, she threw herself headfirst into the industry. She gave herself a self-guided apprenticeship in the field, learning about cleaning chemicals, how to clean a room with maximum efficacy and what to do when something breaks while in a customer’s house.

This trial-by-fire also opened Adora’s eyes to more operational inefficiencies. She submitted dozens of job applications, but only heard back from a handful of companies. Instead of screening and processing applicants, the cleaning companies she applied to manually processed each application and wasted time on back and forth with applicants.

Armed with her apprenticeship, she was ready to shake up the cleaning world.

Systems = success

The business systems that Adora, Aaron and their team have developed demonstrate their engineering ability. This systems approach shows throughout their business, from allowing customers to retain preferred cleaners on their booking platform through to their automated followup of bookings.

Here are some sample questions that job applicants are asked to submit before approval of a real-world cleaning trial:

1. Why do we color-code microfiber cloths?
2. You can spray all-purpose cleaner directly on electronic equipment. True or False?
3. You should mop before you sweep or vacuum. True or False?
4. If a client is not home, it’s okay to use a client’s supplies without their permission. True or False?
5. It is unprofessional to take personal calls while working at a client’s home. True or False?
6. You are at a client’s home and you see a $20 bill on the table. There is no note or message on the table indicating what it is for. There is a wallet on the table. What do you do?
7. You see a pile of unfolded clothes on the floor in your client’s bedroom. What do you do?

I failed, so I’m not going to become a cleaner any time soon.

The quiz doesn’t guarantee quality applicants and doesn’t test attitude or behavior, but the Homejoy recruitment process is still lightyears ahead of other cleaning companies. I’ll argue that HomeJoy’s secret sauce is systematically solving the big, painful problems that hold most cleaning companies back from success. Co-founder Dan talks about pain and pleasure in more depth here.

A personal touch counts


http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
These little ‘touches’ are tested as data points by Homejoy. They are exercises to increase retention, satisfaction and lifetime value. Adora mentioned they are hiring data scientists to delve into the formula for a successful customer experience and measure which personal touches have an influence on a customer returning for another clean.

**Build strategic relationships**

Adora is a Y Combinator alumni and has strong relationships with tech influencers. This worked in her favor while she was trying to raise capital.

Her relationship with Paul Graham paid dividends as he injected capital to keep the company afloat, then later tweeted the graph above, whipping VC firms into a frenzy.

**My review**

Homejoy has grown rapidly and it shows.

My first and second cleaners weren’t up to par, but when I landed on my third cleaner Beverly, I was sold. I wrote out a specific checklist of tasks and set expectations up front. She delivered.

I’ve had shaky early-adopter experiences with a number of San Francisco based ‘service as a service’ companies, but I know from personal experience that training, retaining and motivating great staff is difficult for any startup to execute. We share these struggles in our monthly reports.

Check out some Homejoy reviews here and let me know — would you try Homejoy? Does it solve a problem for you at the right price?

---

**Lecture 4: Why Startups Need to Focus on Sales, Not Marketing**

http://blogs.wsj.com/accelerators/2014/06/03/jessica-livingston-why-startups-need-to-focus-on-sales-not-marketing/

**Jessica Livingston**

The most important thing an early-stage startup should know about marketing is rather counterintuitive: that you probably shouldn’t be doing anything you’d use the term “marketing” to describe. Sales and marketing are two ends of a continuum. At the sales end your outreach is narrow and deep. At the marketing end it is broad and shallow. And for an early stage startup, narrow and deep is what you want — not just in the way you appeal to users, but in the type of product you build. Which means the kind of marketing you should be doing should be indistinguishable from sales: you should be talking to a small number of users who are seriously interested in what you’re making, not a broad audience who are on the whole indifferent.

Successful startups almost always start narrow and deep. Apple started with a computer Steve Wozniak made to impress his friends at the Homebrew Computer Club. There weren’t a lot of them, but they were really interested. Facebook started out just for Harvard University students. Again, not a lot of potential users, but they really wanted it. Successful startups start narrow and deep partly because they don’t have the power to reach a big audience, so they have to choose a very interested one. But also because the product is still being defined. The conversation with initial users is also market research.

See what other startup mentors have to say about marketing tactics.

At Y Combinator, we advise most startups to begin by seeking out some core group of early adopters and then engaging with individual users to convince them to sign up.

For example, the early adopters of Airbnb were hosts and guests in New York City (Y Combinator funded Airbnb in Winter of 2009). To grow, Airbnb needed to get more hosts and also help existing hosts convert better. So Brian Chesky and Joe Gebbia flew to New York every week to meet with hosts — teaching them how to price their listings, take better photos, and so on. They also asked hosts for introductions to potential new hosts, who they then met in person.

Stripe (YC S09) was particularly aggressive about signing up users manually at first. The YC alumni network are a good source of early adopters for a service like Stripe. Co-founders Patrick and John Collison worked their way methodically through it, and when someone agreed to try Stripe, the brothers would install it for them on the spot rather than email a link. We now call their technique “Collison installation.”

---

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Many guest speakers at Y Combinator offer stories about how manual the initial process of getting users was. Pinterest is a mass consumer product, but Ben Silbermann said even he began by recruiting users manually. Ben would literally walk into cafes in Palo Alto and ask random people to try out Pinterest while he gathered feedback over their shoulders.

The danger of the term “marketing” is that it implies the opposite end of the sales/marketing spectrum from the one startups should be focusing on. And just as focusing on the right end has a double benefit — you acquire users and define the product — focusing on the wrong end is doubly dangerous, because you not only fail to grow, but you can remain in denial about your product’s lameness.

All too often, I’ve seen founders build some initially mediocre product, announce it to the world, find that users never show up, and not know what to do next. As well as not getting any users, the startup never gets the feedback it needs to improve the product.

So why wouldn’t all founders start by engaging with users individually? Because it’s hard and demoralizing. Sales gives you a kind of harsh feedback that “marketing” doesn’t. You try to convince someone to use what you’ve built, and they won’t. These conversations are painful, but necessary. I suspect from my experience that founders who want to remain in denial about the inadequacy of their product and/or the difficulty of starting a startup subconsciously prefer the broad and shallow “marketing” approach precisely because they can’t face the work and unpleasant truths they’ll find if they talk to users.

How should you measure if your manual efforts are effective? Focus on growth rate rather than absolute numbers. Then you won’t be dismayed if the absolute numbers are small at first. If you have 20 users, you only need two more this week to grow 10%. And while two users is a small number for most products, 10% a week is a great growth rate. If you keep growing at 10% a week, the absolute numbers will eventually become impressive.

Our advice at Y Combinator is always to make a really good product and go out and get users manually. The two work hand-in-hand: you need to talk individually to early adopters to make a really good product. So focusing on the narrow and deep end of the sales/marketing continuum is not just the most effective way to get users. Your startup will die if you don’t.
5. Business strategy and monopoly
Lecture 5: Business Strategy and Monopoly Theory

Peter Thiel

Sam: Alright, good afternoon, today's speaker is Peter Thiel. Peter was the founder of PayPal, Palantir, and Founders Fund and has invested in most of the tech companies in Silicon Valley. He's going to talk about strategy and competition. Thank you for coming, Peter.

Peter: Awesome, thanks Sam for inviting me, thanks for having me.

I sort of have a single idée fixe that I'm completely obsessed with on the business side which is that if you're starting a company, if you're the founder, entrepreneur starting a company you always want to aim for monopoly and you want to always avoid competition and so hence competition is for losers, something we'll be talking about today.

I'd like to start by saying something about the basic idea of when you start one of these companies, how do we go about creating value. What makes a business valuable? And I want to suggest there's basically a very simple formula, that if you have a valuable company two things are true. Number one, that it creates "X" dollars of value for the world. Number two, that you capture "Y" percent of "X." And the critical thing that I think people always miss in this sort of analysis is that "X" and "Y" are completely independent variables, and so "X" can be very big and "Y" can be very small. "X" can be an intermediate size and if "Y" is reasonably big, you can still have a very big business.

So to create a valuable company you have to basically both create something of value, and capture some fraction of the value of what you've created. And sort of just to illustrate this as a contrast, if you sort of compare the US airline industry with a company like Google on search, if you sort of measure by the size of these industries you could say that airlines are still more important than search, just measured say by revenue. [For airline carriers] there's a hundred ninety five billion in domestic revenues in 2012; Google had just north of fifty billion. And certainly on some intuitive level if you were given a choice and said, well would you want to get rid of all air travel, or do you want to give up search engines, the intuition would be that air travel is something that's more important than search, and this is of course just the domestic numbers. If you'd look at this globally, airlines are much much bigger than search, than Google is, but the profit margins are quite a bit less, they were marginally profitable in 2012, I think in the entire hundred year history of the airline industry, the cumulative profits in the US have been approximately zero. The companies make money, they episodically go bankrupt, they get recapitalized, and you sort of cycle and repeat. And this is reflected in the combined market capitalization of the airline industry may be something like a quarter of Google's. So you have search engine much smaller than air travel but much more valuable. I think this reflects these very different valuations on "X" and "Y."

So if we look at perfect competition, there's some pros and cons to the world of perfect competition. On a high level, this is something you always study in Econ I, it's always it's easy to model, which is why I think Econ Professors like talking about perfect competition. It somehow is efficient, especially in a world where things are static, because you have all the consumer surplus that's captured by everybody, and politically it's what we're told is good in our society: that you want to have competition and that it's somehow a good thing. Of course there are a lot of negatives, it's generally not that good if you're involved in anything that's hyper competitive, because you often don't make money. I'll come back to this a little bit later. So I think on one end of the spectrum you have industries that are perfectly
competitive and at the other end of the spectrum you have things that I would say are monopolies, and they're much stable longer term businesses, you have more capital, and if you get a creative monopoly for inventing something new, I think it's symptomatic of having created something really valuable.

I do think the extreme binary view of the world I always articulate is that there are exactly two kinds of businesses in this world, there are businesses that are perfectly competitive and there are businesses that are monopolies. There is shockingly little that is in between. And this dichotomy is not understood very well because people are constantly lying about the nature of the businesses they are in. And in my mind this is not necessarily the most important thing in business, but I think it's the most important business idea that people don't understand, that there are just these two kinds of businesses.

So let me tell you a little bit about the lies people tell. If you imagine that there was a spectrum of companies from perfect competition to monopoly, the apparent differences are quite small because the people who have monopolies pretend not to. They will basically say, and it's because you don't want to get regulated by the government, you don't want the government to come after you, so you will never say that you have a monopoly. So anyone who has a monopoly will pretend that they are in incredible competition; and on the other end of the spectrum if you are incredibly competitive, and if you're in some sort of business where you will never make any money, you'll be tempted to tell a lie that goes in the other direction, where you will say that you're doing something unique that is somehow less competitive than it looks because you'll want to differentiate, you will want to try and attract capital, or something like that. So if the monopolists pretend not to have monopolies, the non-monopolies pretend to have monopolies, the apparent difference is very small whereas the real difference I would submit is actually quite big. So there's this business distortion that happens because of the lies people tell about their businesses and the lies are sort of in these opposite directions.

Let me drill a little bit down further on the way these lies work. And so the basic lie you tell as a non-monopoly is that we're in a very small market. The basic law is you tell as a monopoly is the market you're in is much bigger than it looks. So typically if you want to think of this in set theoretic terms, you could say that a monopoly tells a lie where you describe your business as the union of these vastly different markets and the non-monopolist describes it as the intersection. So in effect, if you're a non-monopolist you will rhetorically describe your market as super small, you're the only person in that market. If you have a monopoly you'll describe it as super big and there's lots of competition in it.

Some examples of how this works in practice. So I always use restaurants as the example of a terrible business, this is always sort of the idea that capital and competition are antonyms. If someone accumulates capital, a world of perfect competition is a world where all the capital gets competed away. So you're opening a restaurant business, no one wants to invest because you just lose money, so you have to tell some idiosyncratic narrative and you'll say something like, "Well we're the only British food restaurant in Palo Alto." So its British, Palo Alto and of course that's too small a market because people may be able to drive all the way to Mountain View or even Menlo Park and there probably are no people who eat nothing but British food, at least no people who are still alive.

So that's sort of a fictitiously narrow market. There is sort of a Hollywood version of all of this, the way movies always get pitched, where it's like a college football star, you know, joins an elite group of hackers to catch the shark that killed his friend. Now that's a movie that has not yet been made, but the question is, "Is that the right category or is the correct category, it's just another movie?" In which case there are lots of those, it's super competitive, it's incredibly hard to make money, no one ever makes money in Hollywood during movies, it's really really hard.

So you always have this question about is the intersection real? Does it make sense? Does it have value? And of course there are startup versions of this, and in the really bad versions you just take a
whole series of the buzzwords: sharing, mobile, social apps, you combine them and give some kind of narrative and whether or not that's a real business or not, it is generally a bad sign. It's almost this pattern recognition when you have this rhetoric of these sort of intersections, it generally does not work. The something of somewhere is really mostly just the nothing of nowhere, it's like the Stanford of North Dakota, one of a kind, but it's not Stanford.

Let's look at the opposite, the opposite lie, is if you are let's say the search of a company that's down the street from here and has about a happy sixty-six percent market share and is completely dominant in the search market. Google almost never describes itself as a search engine these days and instead it describes itself in all these different ways. So it sometimes says it's an advertising company. So if it was search you would say, well it's like it has this huge market share that's really crazy, so it's like an incredible monopoly, it's a much bigger and much more robust monopoly than Microsoft ever had in the nineties, maybe that's why it's making so much money. But if you say it's an advertising market, you could say well, search advertising is seventeen billion and that's part of online advertising, which is much bigger and then, you know, all US advertising is bigger, and then by the time you get to global advertising, that's close to five hundred billion and so you're talking about three and a half percent, so a tiny part of this much larger market.

Or if you don't want to be an advertising company, you could always say you're a technology company. The technology market is something like a one trillion dollar market and the narrative that you tell about Google and the technology market is, well we're competing with all the car companies with our self-driving cars, we're competing with Apple on TVs and iPhones, we're competing with Facebook, we're competing with Microsoft on office products, we're competing with Amazon on cloud services and so we're in this giant technology market, where there's competition in every direction you look and no we're not the monopoly the government's looking for and we should not get regulated in any way whatsoever. So I think one has to always be super aware that there are these very powerful incentives to distort the nature of these markets, one way or the other.

The evidence of narrow markets in the tech industry is if you basically just, if you look at sort of the big tech companies, Apple, Google, Microsoft, Amazon, they have just been building up cash for year after year and you have these incredibly high profit margins, and I would say that the that one of the reasons the tech industry in the US has been so successful financially is because it's prone to creating all these monopoly-like businesses and it's reflected by the fact that these companies just accumulate so much cash they don't even know what to do with it beyond a certain point.

Let me say a few things about how to build a monopoly, and I think one of the sort of very counterintuitive ideas that comes out of this monopoly thread is that you want to go after small markets. If you're a startup, you want to get to monopoly. You're starting a new company, you want to get to monopoly. Monopolies have a large share of the market, how do you get to a large share of the market? You start with a really small market and you take over the whole market and then over time you find ways to expand that market in concentric circles.

The thing that's always a big mistake is going after a giant market on day one because that's typically evidence that you somehow haven't defined categories correctly, that normally means there is going to be too much competition in one way or another, so I think almost all the successful companies in Silicon Valley had some model of starting with small markets and expanding. If you take Amazon, you start with just a bookstore, we have all the books in the world, it's a better bookstore than anybody else has in the world when it starts in the nineties. It's online, there's things you can do that you could not do before, and then you gradually expand into all sorts of different forms of e-commerce and other things beyond that.
Ebay, you start with Pez dispensers, you move on to Beanie Babies, and eventually it's all these different on auctions for all these sorts of different goods. What's very counterintuitive about many of these companies is they often start with markets that are so small, that people don't think they are valuable at all when you get started. The Pay Pal version of this was we started with power sellers on Ebay, which was about twenty thousand people. When we first saw this happening in December of 1999, January 2000 right after we launched, there was a sense that these were all, it was such a small market, it was terrible, we thought these were terrible customers to have, it's just people selling junk on the internet, why in the world we want to be going after this market.

But there was a way to get a product that was much better for everybody in that market, and we got to something like twenty five, thirty percent market penetration in two or three months, and you've got some walk in to brand recognition, and are able to build the business from there. So I always think these very small markets are quite underrated. The Facebook version of this I always give is that if the initial market at Facebook was ten thousand people at Harvard, it went from zero to sixty percent market share to ten days, that was a very auspicious start. The way this gets analyzed in business schools is always, that's so ridiculous, it's such a small market, it can't have any value at all. So I think the business school analysis of Facebook early on, or of Paypal early on, or of Ebay early on, is that the markets were perhaps so small as to have almost no value and that they would've had little value had they they stayed small, but it turned out there were ways to grow them concentrically and that's what made them so valuable.

There are always these very unique businesses that are doing something that's not been done before and end up having the potential to be a monopoly. If you're the opening line in Anna Karenina, that all happy families are alike and all unhappy families are unhappy in their own special way, the opposite is true in business, where I think all happy companies are different because they're doing something very unique. All unhappy companies are alike because they failed to escape the essential sameness in competition.

One sort of characteristic of a monopoly technology company is some sort of proprietary technology. My sort of crazy, somewhat arbitrary rule of thumb is you want to have a technology that's an order of magnitude better than the next best thing. So Amazon had over ten times as many books, it may not have been that high tech, but you figure oh well it can sell ten times as many books and be more efficient along the way. Pay Pal, Bill Turner was using checks to send money on Ebay, it took seven to ten days to clear, and PayPal could do it more than ten times as fast. You want to have some sort of very powerful improvement in some order of magnitude improvement, on some key dimension. Of course, if you just come with something totally new, it's just like an anthem in providence. I would say the iPhone was the first smart phone that worked, it may not be in fact, but it's so definitely an order of magnitude of an improvement. So I think the technology needs to be designed to give you a massive delta over the next the next best thing.

I think there often are network effects that can kick in and that really help the thing and these are linked to monopolies over time, the thing that is very tricky about network effects is they're often very hard to get started and so you know everyone understands how valuable they are. There's always this incredibly tricky question: why is it valuable to the first person who's doing something. Economies of scale, if you have something with very high fixed costs and very low marginal costs, that's typically a monopoly-like business.

And then there is this thing of branding which is sort of this idea that gets lodged into people's brains. I never quite understand how branding works on site, so I never invest in companies where it's just about branding but it is, I think, a real phenomenon that creates real value. I think one of the things, I'm going to come back to this in a little bit, towards the end, but one of the things that's very striking

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
is that software businesses are often, are for some reason, very good at some of these things. They are especially good at the economies of scale part because the marginal cost of software is zero. So if you get something that works in software it's often significantly better than the existing solution and then you have these tremendous economies of scale and you can scale fairly quickly.

So even if the market starts small, you can grow your business quickly enough to stay at the same size as the growing market and maintain the monopoly of power. Now the critical thing about these monopolies is it's not enough to have a monopoly for just a moment. The critical thing is have one that lasts over time and so in Silicon Valley there is always this sort of idea that you want to be the first mover and I always think in some ways the better framing is you want to be the last mover. You want to be one of the last companies in that category, those are the ones that are really valid. Microsoft was the last operating system, at least for many decades. Google was the last search engine. Facebook will be valuable if it turns out to be the last of social networking site.

And one way to think of this last mover value is this idea that most of the value in these companies exists far in the future. If you do a discounted cash flow analysis of the business, you'll look at all these profit streams, you have a growth rate, the growth rate's much higher that the discount rate and so most of the value exists far in the future. I did this exercise at Pay Pal in March of 2001 we'd been in business for about twenty seven months and the growth rate was a hundred percent a year, we were discounting future cash flows by thirty percent, and it turned out that about three quarters of the value of the business as of 2001 came from cash flows in years 2011 and beyond.

And whenever you do the math on any of these tech companies, you get to an answer that is something like that. So if you are trying to analyze any of the tech companies in Silicon Valley, AirBnB, Twitter, Facebook, any emerging Internet companies, all the ones in Y Combinator, the math tells you that three quarters, eighty-five percent of the value is coming from cash flows in years 2024 and beyond. It's very far in the future and so one of the things that we always overvalue in Silicon Valley is growth rates and we undervalue durability.

Growth is something you can measure in the here and now, you can always track that very precisely. The question of whether a company will be around a decade from now, that's actually what dominates the value equation and that's a much more qualitative sort of a thing. And so if we went back to this idea of these characteristics of monopoly, the proprietary technology, network effects, economies of scale, you can think of these characteristics as ones that exist at a moment in time where you capture a market and take it over but you also want to think about, are these things going to last over time. So there's a time dimension to all these characteristics. So networks in fact often have a great time element where as the network scales, network effects actually get more robust, and so if you have a network business it's often one that can become a bigger and stronger monopoly over time.

Proprietary technology is always a little bit of a tricky one, so you want something that is an order of magnitude better than the state of the art in the world today. That's how you get people's attention, that's how you initially break through. But then you don't want to be superseded by somebody else. So there are all these areas of innovation where there was tremendous innovation, but no one made any money. So disk drive manufacturing in the 1980's, you could build a better disk drive than anybody else, you could take over the whole world and two years later someone else would come along and replace yours. In the course of fifteen years, you got vastly improved disk drives, so had a great benefit to consumers, but it didn't actually help the people who started these companies.

There's always this question about having a huge breakthrough in technology, but then also being able to explain why yours will be the last breakthrough or at least the last breakthrough for a long time or if
you make a breakthrough, then you can keep improving on it at quick enough pace that no one can ever
catch up. So if you have a structure of the future where there's a lot of innovation and other people will
come up with new things and the thing you're working on, that's great for society. It's actually not that
good for your business typically. And then economies of scale we've already about. I think this last
mover thing is very critical. I'm always tempted, I don't want to overdo chess analogies, but the first
mover in chess is someone who plays white, white is about a one-third of a pawn advantage, so there is
a small advantage to going first. You want to be the last mover who wins the game, so there's always
world chess champion Kappa Blanca line, “You must begin by studying the end game.” I wouldn't say
that's the only thing you should study, I think perspective of asking these questions, why will this still
be the leading company in ten, fifteen, twenty years from now, is a really critical one to try to think
through.

I want to go in two slightly other directions with this the monopoly versus competition idea. I think
this is the central idea on my mind for business, for starting a business, for thinking about them. There
are some very interesting perspectives, I think it gives, on the whole history of innovation in
technology and science. We've lived through fifty three hundred years of incredible technological
progress in many many different domains. Steam engines to railways, the telephone, refrigeration,
household appliances, the computer revolution, aviation, all different areas of technological innovation.
Then there's sort of an analogous thing to say about science where we've lived through centuries of
enormous amounts of innovation in science as well.

The thing that I think people always miss when they think about these things, is that because "X" and
"Y" are independent variables, some of these things can be extremely valuable innovations, but the
people who invent them, who come up with them, do not get rewarded for this. Certainly if you go back
to you need to create X dollars in value and you capture Y percent of X, I would suggest that the history
of science has generally been one where Y is zero percent across the board, the scientists never make
any money. They're always deluded into thinking that they live in a just universe that will reward them
for their work and for their inventions. This is probably the fundamental delusion that scientists tend to
suffer from in our society. Even in technology there are sort of many different areas of technology where
there were great innovations that created tremendous value for society, but people did not actually
capture much of the value. So I think there is a whole history of science and technology that can be
told from the perspective of how much value was actually captured. Certainly there are entire sectors
where people didn't capture anything.

You're the smartest physicist of the twentieth century, you come up with special relativity, you come up
with general relativity, you don't get to be a billionaire, you don't even get to be a millionaire. It just
somehow doesn't work that way. The railroads were incredibly valuable, they mostly just went bankrupt
because there was too much competition. Wright brothers, you fly the first plane, you don't make any
money. So I think there is a structure to these industries that's very important.

I think the thing that's actually rare are the success cases. So if you really think about the history in
this and this two hundred fifty years sweep, why is almost always zero percent, it's always zero in
science, it's almost always in technology. It's very rare where people made money. You know in the late
eighteenth, early nineteenth century, the first industrial revolution was the textile mills, you got the
steam engine, you sort of automated things. You had these relentless improvements that people
improved efficiency of textile factories, of manufacturing generally, at a clip of five to seven percent
every year, year after year, decade after decade. You had sixty, seventy years of tremendous
improvement from 1780 to 1850. Even in 1850, most of the wealth in Britain was still held by the
landed aristocracy and the workers didn't make that much. The capitalists didn't make that much either,
it was all competed away. There were hundreds of people running textile factories, it was an industry
where the structure of the competition prevented people from making any money.
Lecture 5: Business Strategy and Monopoly Theory

There are, in my mind, probably only two broad categories in the entire history of the last two hundred and fifty years where people actually came up with new things and made money doing so. One is these sort of vertically integrated complex monopolies which people did build in the second industrial revolution at the end of the nineteenth and start of the twentieth century. This is like Ford, it was the vertically integrated oil companies like Standard Oil, and what these vertically integrated monopolies typically required was a very complex coordination, you’ve got a lot of pieces to fit together in just the right way, and when you assemble that you had a tremendous advantage. This is actually done surprisingly little today and so I think this is sort of a business form that when people can pull it off, is very valuable.

It's typically fairly capital intensive, we live in a culture where it's very hard to get people to buy into anything that's super complicated and takes very long to build. When I think of my college Elon from PayPal success with Tesla and SpaceX, I think the key to these companies was the complex vertically integrated monopoly structure they had. If you look at Tesla or SpaceX and you ask, was there sort of a single breakthrough, I mean they certainly innovated on a law of dimensions, but I don't think there was a single 10X breakthrough on battery storage, they may be working on some things in rocketry, but there was no sort of single massive breakthrough. But what was really impressive was integrating all these pieces together and doing it in a way that was more vertically integrated than most other competitors.

So Tesla also integrated the car distributors so they wouldn't steal all the money as has happened with the rest of the car industry in the US. Or SpaceX, basically, you pulled in all subcontractors where most of the large aerospace companies have single sourced subcontractors that are able to charge monopoly profits and make it very hard for the integrated aerospace companies to make money. And so vertical integration I think is sort of a very under explored modality of technological progress that people would do well to look at more.

And then I think there is something about software itself that's very powerful. Software has these incredible economies of scale, these low marginal costs, and there is something about the world of bits, as opposed to the world of atoms, where you can often get very fast adoption and fast adoption is critical to capturing and taking over markets because even if you have a small market, if adoption rate is too slow, there will be enough time for other people to enter that market and compete with you. Whereas if you have a small to midsize market, have the fast adoption rate, you can now take over this market. So I think this is one of the reasons Silicon Valley has done so well and why software has been this phenomenal industry.

What I would suggest that we will leave you with is there are these different rationalizations people give for why certain things work and why certain things don't work, and I think these rationalizations always obscure this question on creating "X" dollars in value and capturing "Y" percent of "X." So, the science rationalization we're always told is that the scientists aren't interested in making money. They're doing it for charitable reasons and that you're not a good scientist if you're motivated by money. I'm not even saying people should always be motivated by money or something like this, but I think we should wish to be a little bit more critical of this as a rationalization. We should ask is this a rationalization to obscure the fact that "Y" equals zero percent and the scientists are operating in this sort of world where all the innovation is effectively competed away and they can't capture any of it directly.

The software distortion that often happens is because people are making such vast fortunes in software, we infer that this is the most valuable thing in the world being done full stop. And so people at Twitter make billions of dollars, it must be that Twitter is worth far more than anything Einstein did. What that
realization tends to obscure, is again that "X" and "Y" are independent variables and that there are these businesses where you capture a lot of X and others where you don't. So I do think the history of innovation has been this history where the microeconomics, the structure of these industries has mattered a tremendous amount and there is sort of this story where some people made vast fortunes because they worked in industries with the right structure and other people made nothing at all because they were in these very competitive things.

We shouldn't just rationalize that way. I think it's worth understanding this better. Then finally, let me come back to this sort of overarching theme for this talk, this competition is for losers idea, which is always a provocative way to title things because we always think of the losers as the people who are not good at competing. We think of losers as the people who are slow on the track team in high school or do a little less well on standardized tests, and don't get into the right schools. So we always think of losers as people who can't compete and I want us to really rethink and re-value this and consider whether it's possible the competition itself is off.

It's not just the case we don't understand this monopoly competition dichotomy intellectually. So we've been talking about why you wouldn't understand it intellectually, because people lie about it, it's distorted, the history of innovation rationalizes it in all these very strange ways. I think it's more than just an intellectual blind spot, but also a psychological blind spot, where we find ourselves very attracted to competition and in one form or another we find it reassuring if other people do things. The word ape, already in the time of Shakespeare, meant both primate and imitate and that is something about human nature, it's deeply imitative, ape like, sheep like and this is very problematic thing that we need to always think through and try to overcome.

There is always this question about competition as a form of validation, where we go for things that lots of other people are going for and it's not that there is wisdom of crowds, is not that lots of people trying to do something is the best proof of that being valuable, I think it's when lots of people are trying to do something, that is often proof of insanity. There are twenty thousand people a year who move to Los Angeles to become movie stars, about twenty of them make it. I think the Olympics are a little bit better because you have, you can sort of figure out pretty quickly whether you're good or not, so there's little less of a deadweight loss to society. You know the sort of educational experience that at a place, the pre-Stanford educational experience, there is always sort of a non-competitive characterization. I think most of the people in this room had machine guns and they were competing with people bows and arrows, so it wasn't exactly a parallel competition when you were in junior high school, in high school. There is always the question: does the tournament make sense as you keep going?

There is always this question if people going on to grad school or post doctoral educations, does the intensity of the competition really make sense. There is the classic Henry Kissinger line describing his fellow faculty at Harvard, “The battles were so ferocious because the stakes were so small,” describing academia and you sort of think on one level this is a description of insanity. Why would people fight like crazy when the stakes are so small, but it's also, I think, simply a function of the logic of a situation. When it's been really hard to differentiate yourself from other people, when the differences are, when the objectives differences really are small, you have to compete ferociously to maintain a difference of one sort or another. That's often more imaginary than real. There is always sort of a personal version of this I tell, where I was sort of hypertracked. In my eighth grade junior high school yearbook one my friends wrote, I know you're going to get into Stanford. Four years later, I went to Stanford Law School, ended up at a big law firm in New York where from the outside everybody wanted to get in and on the inside everybody wanted to leave and it was this very strange dynamic where I realized, this was maybe not the best idea, and I left after seven months and three days.

http://startupclass.samaltman.com/ (gekregen van Inne ten Haveinne@darwine.nl 06-42804208)
Other people down the hall told me, it's really reassuring to see you leave, Peter, I had no idea that it was possible to escape from Alcatraz, which of course all you had do was go out the front door and not come back. But so much of people's identities got wrapped up in winning these competitions that they somehow lost sight of what was important, what was valuable. Competition does make you better at whatever it is that you're competing at because when you're competing you're comparing yourself with the people around you. I'm figuring out how to beat the people next to me, how do I do somewhat better than whatever it is they're doing and you will get better at that. I'm not questioning that, I'm not denying that, but there often comes this tremendous price that you stop asking some bigger questions about what's truly important and truly valuable. And so I would say, don't always go through the tiny little door that everyone's trying to rush through, maybe go around the corner and go for the best door that nobody is taking, thank you very much.

Q: Do you have any ways to determine the difference [between a monopoly and a non monopoly] when looking at an idea or thinking about your own idea?

A: I would say the question I always focus on is what is the actual market? So not what's the narrative of the market, because you can always tell a fictional story about a market: it's much bigger much or smaller, but what is the real objective market. So you always try to figure it out, and you realize people have incentives to powerfully distort these things.

Q: So which of the aspects, of all these you mentioned, you would you say is applicable to Google?

A: Well, They have they have network effects with the ad network, they had proprietary technology that gave them the initial lead because they had the page rank algorithm, which was an order of magnitude better than any other search engine. They had economies of scale up because of the need to store all these different sites, and at this point you have brand, so Google has all four. Maybe the proprietary technology is somewhat weaker at this point but definitely it had all four, and maybe three and a half out of four now.

Q: How does this apply to Palantir and Square?

A: That's sort of a set of companies that are doing different copycat payment systems, on mobile phones, there's Square, there's PayPal, they have different shapes that's how they differentiate themselves, one is a triangle, one is a square. Maybe at one point the Apes run out of shapes or something like that, but Palantir we started focus with focus on the intelligence community, which is a small submarket, you had a proprietary technology that used a very different approach where it was focused on the human computer synthesis, rather than the substitution, which I think is the dominant paradigm. So, there is a whole set of things I would say on the market approach and on the proprietary technology.

Q: What do you think about lean startups?

A: Yes, of course, so the question is what do I think about lean startups, iterative thinking where you get feedback from people versus complexity that may not work.

I'm personally quite skeptical of all the lean startup methodology. I think the really great companies did something that was somewhat more of a quantum improvement that really differentiated them from everybody else. They typically did not do massive customer surveys, the people who ran these companies sometimes, not always, suffered from mild forms of aspergers, so they were not actually that influenced, not that easily deterred, by what other people told them to do. I do think we're way too focused on iteration as a modality and not enough on trying to have a virtual ESP link with the public.
and figuring it out ourselves. I would say the risk question is always a very tricky one, because it's often the case that you don't have enough time to really mitigate risk. If you're going to take enough time to figure out what people want, you often will have missed the boat by then. And then of course there is always the risk of doing something that's not that significant or meaningful. You could say that a track in law school is a low risk track from one perspective, it may still be a very high risk track in the sense it you maybe you have a high risk of not doing something meaningful with your life. We have to think about risk in these very complicated ways. I think risk is this complicated concept.
Lecture 5: Competition Is for Losers

If you want to create and capture lasting value, look to build a monopoly, writes Peter Thiel

http://online.wsj.com/articles/peter-thiel-competition-is-for-losers-1410535536

Only one thing can allow a business to transcend the daily brute struggle for survival: monopoly profits.

What valuable company is nobody building? This question is harder than it looks, because your company could create a lot of value without becoming very valuable itself. Creating value isn't enough—you also need to capture some of the value you create.

This means that even very big businesses can be bad businesses. For example, U.S. airline companies serve millions of passengers and create hundreds of billions of dollars of value each year. But in 2012, when the average airfare each way was $178, the airlines made only 37 cents per passenger trip. Compare them to Google, GOOGL -0.18% which creates less value but captures far more. Google brought in $50 billion in 2012 (versus $160 billion for the airlines), but it kept 21% of those revenues as profits—more than 100 times the airline industry’s profit margin that year. Google makes so much money that it is now worth three times more than every U.S. airline combined.

The airlines compete with each other, but Google stands alone. Economists use two simplified models to explain the difference: perfect competition and monopoly.

Monopolies are a good thing for society, venture capitalist Peter Thiel argues in an essay on WSJ. The PayPal co-founder joins Sara Murray to discuss his business philosophy, his take on Apple Pay, and what's a deal breaker in pitch meetings.

"Perfect competition" is considered both the ideal and the default state in Economics 101. So-called perfectly competitive markets achieve equilibrium when producer supply meets consumer demand. Every firm in a competitive market is undifferentiated and sells the same homogeneous products. Since no firm has any market power, they must all sell at whatever price the market determines. If there is money to be made, new firms will enter the market, increase supply, drive prices down and thereby eliminate the profits that attracted them in the first place. If too many firms enter the market, they’ll suffer losses, some will fold, and prices will rise back to sustainable levels. Under perfect competition, in the long run no company makes an economic profit.

The opposite of perfect competition is monopoly. Whereas a competitive firm must sell at the market price, a monopoly owns its market, so it can set its own prices. Since it has no competition, it produces at the quantity and price combination that maximizes its profits.

To an economist, every monopoly looks the same, whether it deviously eliminates rivals, secures a license from the state or innovates its way to the top. I'm not interested in illegal bullies or government favorites: By "monopoly," I mean the kind of company that is so good at what it does that no other firm can offer a close substitute. Google is a good example of a company that went from 0 to 1: It hasn’t competed in search since the early 2000s, when it definitively distanced itself from Microsoft MSFT -0.28% and Yahoo!

Americans mythologize competition and credit it with saving us from socialist bread lines. Actually, capitalism and competition are opposites. Capitalism is premised on the accumulation of capital, but under perfect competition, all profits get competed away. The lesson for entrepreneurs is clear: If you want to create and capture lasting value, don't build an undifferentiated commodity business.

How much of the world is actually monopolistic? How much is truly competitive? It is hard to say because our common conversation about these matters is so confused. To the outside observer, all businesses can seem reasonably alike, so it is easy to perceive only small differences between them. But the reality is much more binary than that. There is an enormous difference between perfect competition and monopoly, and most businesses are much closer to one extreme than we commonly realize.

The confusion comes from a universal bias for describing market conditions in self-serving ways: Both monopolists and competitors are incentivized to bend the truth.

Monopolists lie to protect themselves. They know that bragging about their great monopoly invites being audited, scrutinized and attacked. Since they very much want their monopoly profits to continue unmolested, they tend to do whatever they can to conceal their monopoly—usually by exaggerating the power of their nonexistent competition.

Google makes so much money that it is now worth three times more than every U.S. airline combined.

Think about how Google talks about its business. It certainly doesn’t claim to be a monopoly. But is it one? Well, it depends: A monopoly in what? Let’s say that Google is primarily a search engine. As of May 2014, it owns about 68% of the search market. (Its closest competitors, Microsoft and Yahoo!, YHOO +0.34% have about 19% and 10%, respectively.) If that doesn’t seem dominant enough, consider the fact that the word “google” is now an official entry in the Oxford English Dictionary—as a verb. Don’t hold your breath waiting for that to happen to Bing.

But suppose we say that Google is primarily an advertising company. That changes things. The U.S. search-engine advertising market is $17 billion annually. Online advertising
is $37 billion annually. The entire U.S. advertising market is $150 billion. And global advertising is a $495 billion market. So even if Google completely monopolized U.S. search-engine advertising, it would own just 3.4% of the global advertising market. From this angle, Google looks like a small player in a competitive world.

What if we frame Google as a multifaceted technology company instead? This seems reasonable enough; in addition to its search engine, Google makes dozens of other software products, not to mention robotic cars, Android phones and wearable computers. But 95% of Google’s revenue comes from search advertising; its other products generated just $2.35 billion in 2012 and its consumer-tech products a mere fraction of that. Since consumer tech is a $964 billion market globally, Google owns less than 0.24% of it—a far cry from relevance, let alone monopoly. Framing itself as just another tech company allows Google to escape all sorts of unwanted attention.

Non-monopolists tell the opposite lie: "We're in a league of our own." Entrepreneurs are always biased to understated scale of competition, but that is the biggest mistake a startup can make. The fatal temptation is to describe your market extremely narrowly so that you dominate it by definition.

Suppose you want to start a restaurant in Palo Alto that serves British food. "No one else is doing it," you might reason. "We'll own the entire market." But that is only true if the relevant market is the market for British food specifically. What if the actual market is the Palo Alto restaurant market in general? And what if all the restaurants in nearby towns are part of the relevant market as well?

These are hard questions, but the bigger problem is that you have an incentive not to ask them at all. When you hear that most new restaurants fail within one or two years, your instinct will be to come up with a story about how yours is different. You'll spend time trying to convince people that you are exceptional instead of seriously considering whether that is true. It would be better to pause and consider whether there are people in Palo Alto who would rather eat British food above all else. They may well not exist.

In 2001, my co-workers at PayPal and I would often get lunch on Castro Street in Mountain View, Calif. We had our pick of restaurants, starting with obvious categories like Indian, sushi and burgers. There were more options once we settled on a type: North Indian or South Indian, cheaper or fancier, and so on. In contrast to the competitive local restaurant market, PayPal was then the only email-based payments company in the world. We employed fewer people than the restaurants on Castro Street did, but our business was much more valuable than all those restaurants combined. Starting a new South Indian restaurant is a really hard way to make money. If you lose sight of competitive reality and focus on trivial differentiating factors—maybe you think your naan is superior because of your great-grandmother's recipe—your business is unlikely to survive.

The problem with a competitive business goes beyond lack of profits. Imagine you're running one of those restaurants in Mountain View. You're not that different from dozens of your competitors, so you've got to fight hard to survive. If you offer affordable food with low margins, you can probably pay employees only minimum wage. And you'll need to squeeze out every efficiency: That is why small restaurants put Grandma to work at the register and make the kids wash dishes in the back.

A monopoly like Google is different. Since it doesn't have to worry about competing with anyone, it has wider latitude to care about its workers, its products and its impact on the wider world. Google's motto—"Don't be evil"—is in part a branding ploy, but it is also characteristic of a kind of business that is successful enough to take ethics seriously without jeopardizing its own existence. In business, money is either an important thing or it is everything. Monopolists can afford to think about things other than making money; non-monopolists can't. In perfect competition, a business is so focused on today's margins that it can't possibly plan for a long-term future. Only one thing can allow a business to transcend the daily brute struggle for survival: monopoly profits.

So a monopoly is good for everyone on the inside, but what about everyone on the outside? Do outsized profits come at the expense of the rest of society? Actually, yes: Profits come out of customers' wallets, and monopolies deserve their bad reputation—but only in a world where nothing changes.

In a static world, a monopolist is just a rent collector. If you corner the market for something, you can jack up the price; others will have no choice but to buy from you. Think of the famous board game: Deeds are shuffled around from player to player, but the board never changes. There is no way to win by inventing a better kind of real-estate development. The relative values of the properties are fixed for all time, so all you can do is try to buy them up.

But the world we live in is dynamic: We can invent new and better things. Creative monopolists give customers more choices by adding entirely new categories of abundance to the world. Creative monopolies aren't just good for the rest of society; they're powerful engines for making it better.

Even the government knows this: That is why one of its departments works hard to create monopolies (by granting patents to new inventions) even though another part hunts them down (by prosecuting antitrust cases). It is possible to question whether anyone should really be awarded a monopoly simply for having been the first to think of something like a mobile software design. But something like Apple's AAPL -0.86%monopoly profits from designing, producing and marketing the iPhone were clearly the reward...
for creating greater abundance, not artificial scarcity. Customers were happy to finally have the choice of paying high prices to get a smartphone that actually works. The dynamism of new monopolies itself explains why old monopolies don't strangle innovation. With Apple's iOS at the forefront, the rise of mobile computing has dramatically reduced Microsoft's decadeslong operating system dominance.

Before that, IBM's IBM -0.50% hardware monopoly of the 1960s and '70s was overtaken by Microsoft's software monopoly. AT&T T -0.53% had a monopoly on telephone service for most of the 20th century, but now anyone can get a cheap cellphone plan from any number of providers. If the tendency of monopoly businesses was to hold back progress, they would be dangerous, and we'd be right to oppose them. But the history of progress is a history of better monopoly businesses replacing incumbents. Monopolies drive progress because the promise of years or even decades of monopoly profits provides a powerful incentive to innovate. Then monopolies can keep innovating because profits enable them to make the long-term plans and finance the ambitious research projects that firms locked in competition can't dream of.

So why are economists obsessed with competition as an ideal state? It is a relic of history. Economists copied their mathematics from the work of 19th-century physicists: They see individuals and businesses as interchangeable atoms, not as unique creators. Their theories describe an equilibrium state of perfect competition because that is what's easy to model, not because it represents the best of business. But the long-run equilibrium predicted by 19th-century physics was a state in which all energy is evenly distributed and everything comes to rest—also known as the heat death of the universe. Whatever your views on thermodynamics, it is a powerful metaphor. In business, equilibrium means stasis, and stasis means death. If your industry is in a competitive equilibrium, the death of your business won't matter to the world; some other undifferentiated competitor will always be ready to take your place.

Perfect equilibrium may describe the void that is most of the universe. It may even characterize many businesses. But every new creation takes place far from equilibrium. In the real world outside economic theory, every business is successful exactly to the extent that it does something others cannot. Monopoly is therefore not a pathology or an exception. Monopoly is the condition of every successful business.

Tolstoy famously opens "Anna Karenina" by observing: "All happy families are alike; each unhappy family is unhappy in its own way." Business is the opposite. All happy companies are different: Each one earns a monopoly by solving a unique problem. All failed companies are the same: They failed to escape competition.

Adapted from Mr. Thiel’s new book, with Blake Masters, "Zero to One: Notes on Startups, or How to Build the Future," which will be published by Crown Business on Sept. 16. Mr. Thiel co-founded PayPal and Palantir and made the first outside investment in Facebook.
As the title suggests, it’s no roundup of feel-good stories about passion and vision. Instead, Horowitz, who is co-founder of the VC firm Andreessen Horowitz and the software company Loudcloud (later Ops-ware), offers blunt advice on surviving the rough stuff startup life dishes out. He spoke recently with Inc. editor-at-large Leigh Buchanan.

What you call “the struggle” of startup life will either kill a CEO or make him stronger. How can you improve the odds of that second outcome?

A lot of being a successful CEO is not quitting. It sounds almost corny. Winners never quit and quitters never win. But there’s something deeper. When you’re in it, you are building a tremendous amount of knowledge about the company, about the market, about the customer, and about the product. The longer you have to apply that, the better your chances are. If you can somehow stay in the box, then, amazingly, you might find yourself in a good position.

You advocate getting bad news out quickly. In the fragile early stages of a company, how do you do that without destroying morale?

As CEO, you feel as if you’re the daddy who has got to protect the children from bad news. Because if it hurts you, then it is going to devastate them. The opposite is true. Employees are not nearly as married to the company as you are. They do not take the losses as hard. And they want the chance to work on solving those problems. Culturally, once you deliver bad news a few times, people get over the worrying part. Employees think, Every week, Ben is telling us something else that’s wrong with the company. He tells us, we fix it, and that’s great.

In all kinds of lousy situations—including firing people—you want the CEO to accept that the failure is her own. Why is that so important?

Because it’s true. To get a job as an executive in a company, someone has to have tremendous credentials coming in. I’ve never fired executives because they didn’t work hard or they weren’t smart. The issue is always that they had the wrong background and the wrong skill set for the company at that point in time. Whose fault is that? You’re the one who knew your company. And you’re the one who got the chance to assess the candidate. You got it wrong. Or else you failed to integrate the person. [Accepting the failure as your own] also puts you in a much more fair position in terms of the message you deliver to the person being fired. As my friend [Intuit chairman] Bill Campbell says, when you fire people, you have to take their job but you don’t have to take their dignity.

What forces encountered in the startup crucible work against your ability to create a great place to work? How can you counteract them?

One of the biggest things that can work against a startup being good is growth. A lot of what makes a company good is common knowledge. If everybody in a company knows everything, then generally it is going to be a pretty good place to work. Communication is super high fidelity. Everybody is on the same page, and 99 percent of the work people do actually impacts the company. The bigger you get, the more siloed information gets, the less common knowledge you have, and the more likely people are working on something that ultimately the company cares nothing about. There’s politics and backbiting and nonsense. The challenge for the CEO is to make sure that as the company gets bigger, there is a good communication architecture. That’s a really hard thing.

For most companies, culture evolves over time. But you advocate creating “cultural design points” that strongly influence people’s behaviors. How do you do that?

You’ve got to think about things that are both important to the business and that differentiate you. When we started Andreessen Horowitz, we knew that one of the most frustrating things for entrepreneurs is that VCs do not respect their time. As entrepreneurs, every time we’d visit a VC, we’d wait in the lobby for 30 to 45 minutes. So at the core of the firm, we wanted a cultural tenet to be respect for the entrepreneur and the entrepreneurial process.

But how do you get that into people’s minds? The mechanism I came up with was to tell VCs that if you are late for a meeting with an entrepreneur, then the fine is $10 a minute. That is a very big fine. It is shocking to people who have to pay it. And that’s the point. Because every time somebody pays a fine, we get to tell the story of why the person is paying so much money. It really locks it in.

What do you like about difficult decisions?

There are the decisions that you think are a good idea and everybody else agrees. Those are easy. Then there are decisions that you think are a good idea and most people think are a bad idea. If you go with the crowd and get it wrong, then everyone will forgive you because none of them saw it coming. But if you do what you think is right and
you’re wrong, then everyone is going to know you are an idiot. So it’s hard to make the decisions that nobody else likes. When Larry Page decided to focus on artificial intelligence, people thought it was the craziest thing in the world, but it ended up creating the technology for an $8 billion business. That’s the harsh reality of the job. You have to do what you think is right. If you don’t, you are not a leader.

Bad advice from VCs or veteran entrepreneurs can make hard things even harder. Ben Horowitz offers three examples of conventional wisdom that’s not so wise:

**The advice:** Bring in big-company executives early.

**The problem:** The job of a big-company executive is very different from the job of a small-company executive. That creates a potential mismatch that prevents the hire from contributing immediately.

**The advice:** Strive for consensus on executive hiring decisions.

**The problem:** The opinions of people who have not done a particular job may not be relevant to identifying a world-class hire.

**The advice:** Match competing offers for key employees.

**The problem:** Word gets out, and soon everyone thinks the best way to get a raise is to threaten to quit.
6. Growth
Lecture 6: Growth

Alex Schultz

This is awesome; I’ve been watching the lectures on this course, and isn’t it absolutely amazing, the content? And now, you’re stuck with me. We’ll see how that goes.

Unlike Paul when he was talking in the Q&A and you guys asked him what he’d do if he was in college today and he said physics, I actually indulged myself. I went and did Physics at Cambridge. I think physics is an amazing class to give you transferable skills that are really useful in other areas, but that’s not why you’re listening to me today; physics isn’t the class.

So I paid for college doing online marketing, directions marketing. I started with SEO in the 1990s. I created a paper airplane site, and had a monopoly in the small niche market of paper airplanes. When you want to start a startup also see how big the market could be. (In the long term, it wasn’t great.) But what that taught me was how to do SEO. And back in those days it was Alta Vista, and the way to do SEO was to have white text, on a white background, five pages below the fold, and you would rank top of Alta Vista if you just said planes 20 or 30 times in that text. And that was how you won SEO in the 1990s. It was a really, really easy skill to learn.

When I went to college, being a Physicist, I thought paper airplanes would make me cool. I was actually the nerdiest person in the physics class, so I created a cocktail site, which was how I learned to program and that grew to be the largest cocktail site in the UK. That really got me into SEO properly when Google launched. So with Google you had to worry about page rank and getting links back to your site, which basically at that stage meant one link from the Yahoo directory, got you to the top listing in Google if you had white text and a white background below the fold as well.

When Google launched AdWords, that’s really when I started to do all my marketing. That meant buying paid clicks from Google and reselling them to eBay for a small margin of like 20% using their affiliate program. That was what really kicked me into over-drive into doing what everyone nowadays talks about as growth - growth hacking or growth marketing. In my mind it’s just internet marketing using whatever channel you can to get whatever output you want, and that’s how I paid for college and that’s how I went from being a Physicists to a Marketer - transitioning to the darkside of the force.

So what do you think matters most for growth? You’ve had tons of lectures, and people have said it over and over, so what do you guys think matters most for growth?

Audience: Great product.
Schultz: What does a great product lead to?
Audience: Customers.
Schultz: And what do you need those customers to do?
Audience: Spread the word.

Schultz: Yes that’s it, retention. Retention is the single most important thing for growth. Now we have an awesome growth team at Facebook and I’m super proud to work in it, but the truth of the matter is, we have a fantastic product. Getting to work on growth of Facebook is a massive privilege because we are promoting something that everyone in the world really wants to use, which is absolutely incredible. If we can get people on, and get them ramped up, they stick on Facebook.
So many times, I got to advise multiple start-ups. My favorite was working with Airbnb, but I've worked with Coursera, I've worked with other ones that haven't done as well as those guys. But the one thing that's true, over and over again is, if you look at this curve, 'percent monthly active' versus 'number of days from acquisition', if you end up with a retention curve that asymptotes to a line parallel to the X-axis, you have a viable business and you have product market fit for some subset of market. But most of the companies that you see fly up, we've talked about packing and virality and all of this stuff, their retention curve slopes down toward the axis, and in the end, intercepts the X-axis.

Now when I show this job to people, they say that’s all well and good, you had a million people a day in terms of growth, when you started the growth team at Facebook, or 'you were at 50 million users, you had a lot of people join your site so you had a ton of data to do this.' We use the same methodology for our B2B growth, getting people to sign-up for services advertisements, we use this to understand how much growth we were going to have in that market as well. And at that point when I joined Facebook, the product was three days old. And within 90 days of the product launching, we were able to use this technique to figure out what the one year value of an advertiser was, and we predicted it for the first year to 97%. So I think it's very important to look at your retention curve.

If you see here, this red line is the ‘number of users’ who have been on your product for a certain number of days. So a bunch of people, will have been on the product at least one day, but if your product has been around for a year, you’ll have zero users who have been on it for 366 days. Make sense?

So what you then do is look for all of your users who have then been on your product one day. What percentage of them are monthly active? 100% for the first 30 days obviously, because monthly active, they also end up on one day. But then you look at 31. Every single user on their 31st day after registration, what percentage of them are monthly active? Thirty-second day, thirty-third day, thirty-fourth day. And that allows you, with only 10,000 customers, to get a real idea of what this curve is going to look like for your product. And you're going to be able to tell, does it asymptote? It'll get noisy towards the right side, like I'm not using real data, but you'll be able to get a handle on, whether this curve flattens out or does it not. If it doesn’t flatten out, don’t go into growth tactics, don’t do virality, don’t hire a growth hacker. Focus on getting product market fit, because in the end, as Sam said in the beginning of this course: idea, product, team, execution. If you don't have a great product, there's no point in executing more on growing it because it won't grow. Number one problem I've seen, inside Facebook for new products, number one problem I've seen for startups, is they don't actually have product market fit, when they think they do.

So the next question that people ask over and over again is, what does good retention look like? Sure! I can have 5% retention, but I'm guessing Facebook had better than that. That's not going to be a successful business. I get really pissed off when people ask me that question, because I think you can figure it out. I love this story; this is like my one gratuitous story (link on powerpoint) that I'm throwing out here, so the rest of it may not be as gratuitous. But this is a picture that was published in Life Magazine in 1950 of one of the Trinity nuclear bomb tests. There's a guy named Jeffrey Taylor. He was a British Physicist who ended up winning the Nobel Prize. He was able to figure it out, from looking at this picture (picture on powerpoint) what the power of the U.S. atomic bomb was, and Russians were publishing similar pictures, just using dimensional reasoning. Dimensional reasoning was one of the best skills I learned during my time studying physics back in the UK. What dimensional reasoning is, you look at the dimensions that are involved in a problem, so you want to figure out energy, newtons, meters, newtons as a kilogram, meters seconds to minus two. You want to figure out kilograms, meters squared, seconds to minus two, and then you try to figure out how you can get each of those numbers from what data you have. The mass is the volume of this sphere, so that’s a meter cubed, so you've got
meters to five over seconds to minus two and he was able to use that to figure out what the power of this atomic bomb was and what the ratios of the power between the Russian and the U.S. atomic bomb was, and essentially reveal one of the top secrets that existed in the world at that time.

That's a hard problem. Figuring out what Facebook's retention rate is, is not a hard problem. How many people are there on the internet? 2.4 billion, 2.3 billion. Okay, Facebook is banned in China, so what now?

Audience: 2 billion.

Schultz: So 2 billion people on the internet. Facebook said around 1.3 billion users in terms of active users. You can divide those numbers by each other. And yet that won't give you the right answer. Of course not! But it's going to give you close enough to a ballpark answer of what the retention rate looks like for Facebook. If we signed everyone on the internet up, then you will know it's higher than that. Similarly, if you look at WhatsApp. They've announced 600 million active users. How many people have Smart Phones? You can figure out that number - that number is out knocking around. It can give you an idea of how many users there are. Amazon has a had a pop at signing up almost everyone in the United States. You know how many people are online in the United States, and you have a good idea of how many customers Amazon has from the numbers they throw out. Different verticals need different terminal retention rates for them to have successful businesses. If you're on ecommerce and you're retaining on a monthly active basis, like 20 to 30% of your users, you're going to do very well. If you're on social media, and the first batch of people signing up to your product are not like, 80% retained, you're not going to have a massive social media site. So it really depends on the vertical you're in, what the retention rates are. What you need to do is have the tools to think, 'who out there is comparable' and how you can look at it and say, 'am I anywhere close to what real success looks like in this vertical?'

Retention is the single most important thing for growth and retention comes from having a great idea and a great product to back up that idea, and great product market fit. The way we look at, whether a product has great retention or not, is whether or not the users who install it, actually stay on it long-term, when you normalize on a cohort basis, and I think that's a really good methodology for looking at your product and say 'okay the first 100, the first 1,000, the first 10,000 people I get on this, will they be retained in the long-run?

So now, how do you attack operating for growth? Let's say you have awesome product market fit. You've built an ecommerce site, and you have 60% of people coming back every single month, and making a purchase from you, which would be absolutely fantastic. How do you then take that, and say, 'now it's time to scale.' (Now it's time to execute was the last thing on your forum right? *to moderator*. ) That's where I think growth teams come in.

My contrarian viewpoint or whatever, is if you're a startup, you shouldn't have a growth team. Startups should not have growth teams. The whole company should be the growth team. The CEO should be the head of growth. You need someone to set a North star for you about where the company wants to go, and that person needs to be the person leading the company, from my opinion, that's what I've seen. Mark is a fantastic example of that. Back when Facebook started, a lot of people were putting out their registered user numbers. Right? You'd see you registered user numbers for MySpace, you'd see a registered user numbers for __11:38, you'd see registered user numbers. Mark put out monthly active users, as the number both internally he held everyone to, and said we need everyone on Facebook, but that means everyone active on Facebook, not everyone signed up on Facebook, so monthly active people was the number internally, and it was also the number he published externally. It was the number he made the whole world hold Facebook to, as a number that we cared about. If you look at

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
what Jan has done with WhatsApp I think that's another great example. He always published sends numbers.

If you're a Messaging Application, sends is problem the single most important number. If people use you once a day, maybe that's great, but you're not really their primary messaging mechanism, so Jan published the sends number. Inside Airbnb, they talk about ‘nights booked’ and also published that in all of the infographics you see in side TechCrunch. They always benchmark themselves against how many nights booked they have compared to the largest hotel chains in the world. They have at each of these companies, a different North star. The North star doesn't have to be the number of active users for every different vertical. For eBay, it was Gross Merchandise Volume. How much stuff did people actually buy through eBay? Everyone externally tends to judge eBay based on revenue. Actually, Benedict Evans has done this amazing breakdown of Amazon's business, which is really interesting to look at their marketplace business versus their direct business. So eBay is all marketplace business, right? So eBay's being judged by its revenue, when it actually has 10 times Gross Merchandise Volume going through the site. That was the number that eBay looked at when I was working there. Every different company when it thinks about growth, needs a different North star; however, when you are operating for growth it is critical that you have that North star, and you define as a leader.

The reason this matters is, the second you have more than one person working on something, you cannot control what everyone else is doing. I promise you, having now hit 100 people I'm managing, I have no control. It's all influence. It's like I tell one person to do one thing, but the other 99 are going to do whatever they want. And the thing is, it's not clear to everybody what the most important thing is for a company. It would be very easy for people inside eBay to say, 'you know what? we should focus on revenue,' or 'we should focus on the number of people buying from us' or 'we should focus on how many people list items on eBay.' And Pierre, and Meg, and John, those guys as various leaders, have always said 'no, its the amount of Gross Merchandise Volume that goes through our site, the percentage of e-commerce that goes through our site, that is what really matters for this company. This means that when someone is having a conversation and you’re not in the room, or when they’re sitting in front of their computer screens, and thinking about how they built this particular project or this particular feature, in their head it's going to be clear to them that it's not about revenue, it's about Gross Merchandise Volume, or it's not about getting more registrations, registrations don’t matter, unless they become long-term active users. A great example of this was when I was at eBay in 2004, we changed they way we paid our affiliates for new users. Affiliate programs are a bit out of fashion these days, but the idea of an affiliate program is essentially, you pay anyone on the internet a referral for sending traffic to your site, but it’s mostly about getting access to big marketers who do it on their own.

We were paying for confirmed registered users, so all of our affiliates were lined up on getting confirmed registered users to the eBay site. We changed our payment model to pay for activated confirmed registered users. So you had to confirm your account and then bid on an item, or buy or list an item, to become someone that we paid for. Overnight when we made that change, we lost something like 20% of confirmed registered users that were being driven by the affiliates. But the ACRUs (15:45) only dropped by about 5%. The ratio between CIU to ACRU went up, and then, the growth of ACRUs massively accelerated. The cause of this is, if you want to drive CIU, if someone searches for a trampoline, you land them on the registration page because they link you have to register and confirm before they get their trampoline. If you want to drive ACRUs, you land them on the search results page, within eBay for trampolines, so they can see the thing they want to buy, get excited, and then register when they want to buy it. And if you drive just CIUs, people don’t have an amazing magic moment on eBay, when they visit the site. And that’s the next most important thing to think about: How do you drive to the magic moment that gets people hooked on your service.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
In the lecture notes for this course, I’ve stuck in a bunch of links to people I think are brilliant at this stuff. For example, regarding the retention curve I showed you earlier, there’s a link to this guy Danny Ferante who is incredible talking about retention curves. The magic moment there are two videos linked: one is Chamath talking about growth, who is the guy who set up the growth team at Facebook, and the other, is my friend Naomi and I talking at F8 four years about how we were thinking about growth back then. In both of those videos, we talk about the magic moment. What do you think the magic moment is for when you’re signing up to Facebook?

Audience: See your friends.

Schultz: See your friends. Simple as that. I’ve talked to so many companies, and they try to get incredibly complicated about what they’re doing, but it is just as simple as when you see the first picture of one of your friends on Facebook, you go ‘Oh my God, this is what this site is about!’ Zuckerberg talked at Y Combinator about getting people to tend friends in 14 days; that is why we focus on this metric. The number one most important thing in a social media site is connecting to your friends, because without that, you have a completely empty newsfeed, and clearly you’re not going to come back; you’ll never get any notifications, and you’ll never get any friends telling you about things they are missing on the site.

So for Facebook the magic moment, is that moment when you see your friend’s face, and everything we do on growth, if you look at the Linkedin registration flow, you look at the Twitter registration flow, or you look at what WhatsApp does when you sign up, the number one thing all these services look to do, is show you the people you want to follow, connect to, send messages to, as quickly as possible, because in this vertical, this is what matters. When you think about Airbnb or eBay, it’s about finding that unique item, that PEZ dispenser or broken laser pointer, that you really really cared about and want to get ahold of. Like when you see that collectible that you are missing, that is the real magic moment on eBay. When you look on Airbnb and you find that first listing, that cool house you can stay in, and when you go through the door, that’s a magic moment. Similarly on the other side, when you’re listing your house, that first time you get paid, is your magic moment or when you list an item on eBay, the first time you get paid, is your magic moment. You should ask Brian what he thinks, because they’ve done these amazing story boards which I think has been shared, about the journey through a user’s life on Airbnb and how exciting it is. He’ll be talking in around three lectures time; he’s awesome about talking about the magic moment, and getting users to feel the love, joy, and all this stuff.

Think about what the magic moment is for your product, and get people connected to it as fast as possible, because then you can move up where that blue line has asymptotic, and you can go from 60% retention to 70% retention easily if you can connect people with what makes them stick on your site.

The second thing to think about, that everyone in the Valley gets wrong is, we optimize when we think about growth for ourselves. My favorite example is notifications. Again, I’ve talked to and advised many different companies; every single company when they talk about notifications goes ‘Oh, I’m getting too many notifications, I think that’s what we have to optimize for on notifications.’ Okay, are your power users leaving your site because they’re getting too many notifications? No. Then why would you optimize that? They’re probably grown-ups and they can use filters.

What you need to focus on is the marginal user. The one person who doesn’t get a notification in a given day, month, or year. Building an awesome product is all about think about the power user, right? Building an incredible product is definitely optimizing it for the people who use your product the most, but when it comes to driving growth, people who are already using your product are not the ones you have to worry about. So in this Danny Ferante video there’s also talk about our growth accounting framework that we use to think about for growth. We looked at new users, resurrected users (people
who weren’t on Facebook for 30 days and came back) and churned users. The resurrected and churned numbers for pretty much every product I’ve ever seen dominate the new user account once you reach a sensible point of growth a few years in. And all those users who are churning and resurrecting, had low friend counts, and didn’t find their friends so weren’t connected to the great stuff that was going on on Facebook. So the number one thing we needed to focus on, was getting them to those 10 friends, or whatever number of friends they needed. So think about the user on the margin; don’t think about where yourself (21:08), when you’re thinking about growth.

So for operating for growth, what you really need to think about, is what is the North star of your company: What is that one metric, where if everyone in your company is thinking about it and driving their product towards that metric and their actions towards moving that metric up, you know in the long-run your company will be successful. By the way, they’re all probably all correlated to each other, so it’s probably fine to pick almost any metric, whichever one you feel the best about, that aligns with your mission and your values - probably go for that one. But realistically, daily active users fairly correlate to monthly active users; we could have gone with either one. Amount of content shared, also correlates with how many users there are, because guess what? You add a user, they share content. So a lot of things end up being correlate. Pick the one that fits with you and know that you’re going to stick with for a long time. Just have a North Star, and know the magic moment that you know when a user experiences that, they will deliver on that metric for you on the North Star, and then think about the marginal user, don’t think about yourself. Those are, I think, the most important points when operating for growth. Everything has to come from the top.

So the last area is tactics. (*pointing to PowerPoint slide* This is great guy by the way, Tom Fishburne, he lives in the bay area and does these really cynical cartoons I love. → should this be included?) So let’s say you’ve found your niche market that you’re going to have a monopoly on inside the mousetrap market. It’s a silenced mousetrap fir sitting under beds, so if that the mice come to your bed overnight, they can be killed without waking you up. That’s your niche market. Your mousetrap is better than anybody else for that market. What typically happens in Silicon Valley is, everyone thinks marketers are useless. I thought marketers were useless when I was a Physics student, so I’m sure that you guys as Engineering students you must think that we’re awful people who aren’t useful to have around.

‘Build it and they will come.’ That is something that is very much the mantra in the Valley, and I don’t believe it’s true; I believe you actually have to work. There’s a good article in the lecture page from interviewing Ben Silverman. We talked about how the growth of Pinterest was driven by marketing. I’m biased of course.

The first topic I want to talk about is internationalization. Facebook internationalized too late. Sheryl said it broadly in public and I definitely agree with that.

One of the biggest barriers to our long-term growth, and one of the biggest things we had to deal with, was all the countries where there were clones. Famously ? (23:55) had Fakebook.css in their HTML, and there were a ton of sites like that out there, whether it was ?, a clear clone, Mixie, Cyworld, Orkut; they were all these different social networks around the world that grew up when Facebook was focused around the U.S. Internationalizing was an important barrier we needed to knock down, and knocking down barriers is often important to think about for growth. Facebook started out as college-only, so every college that it was launched in was knocking down a barrier. When Facebook expanded beyond colleges to high schools, I wasn’t at the company, but that was a company-shaking moment where people questioned whether or not Facebook would survive, if the culture of the site could survive.
Then after, expanding from high schools to everyone - that was just before I joined - it was a shocking moment; that’s what spurred the growth up to 50 million, and then we hit a brick wall. When we hit that brick wall, that was when a lot of existential questions were being asked inside Facebook whether any social network could ever get to more than 100 million users. It sounds stupid now, but at that time, no one had ever achieved it. Everyone had tapped out between 50 and 100 million users, and we were worried that it wasn’t possible. That was the point at which the growth team got set up; Chamath brought a bunch of us together. He said very publicly he wanted to fire me on multiple occasions. Without Chamath, I think none of us would have stayed at the company; we were a really weird bunch of people - but it worked out. The two things we did, I think that really drove growth initially was, 1) We focused on that 10 friends in 14 days 2) Getting users to the magic moment. That was something that Zuck drove because we were all stuck in analysis paralysis and, ‘Is it causation? Is it correlation?’ Zuck would say ‘You really think that if no one gets a friend, that they'll be active on Facebook? Are you crazy?’

The second thing was internationalization - knocking down another barrier. When we launched it, I think there were two things we did really well: 1) Even though we were late (and stressed about being late) we took the time to build it in a scalable way; we moved slow to move fast. You can actually view the full story from Naomi on one of the video links from the lecture page. What we did was draw all the strings on the site in FBT, which is our translation extraction script and then, we created the community translation platform, so we didn’t have just professional translators translating the site, but we could have all our users translating the site. We got French translates in 12 hours. We managed to get, to this day, 104 languages translated by Facebook for Facebook, 70 of those are translated by the community. We took the time to build something, that would enable us to scale.

The other thing is that we prioritized the right languages. Back then, the four main languages were French, Italian, German, and Spanish (and Chinese, but we are blocked in China). Now look at that list - that’s today’s distribution of languages. Italian isn’t on the list anymore; French and German are about to fall off. In the last year we quadrupled the number of people on Facebook in Hindi. Building for what the world is today is an easy mistake to make, and it’s a lot of what the other social networks did. We built a scalable translation infrastructure that actually enabled us to attack all of the languages, so we could be ready for where the future is going to be. You’ll probably be able to see some of our Internet.org summit in India about where we want to go with language translations.

These are the tactics I want to go through now: Virality, SEO, ESPN, SEM, Affiliates/referral programs. I think there are two ways to look at virality. There’s a great book by Adam L. Penenberg called the Viral Loop that goes through a bunch of case studies of companies that have grown through viral marketing. I strongly encourage you to read this book if you’re interested in viral marketing, as well as advertising. He has some really great creative tips. So virality. Chamath has this really great model that he told us about when I joined Facebook, which is to think about virality about a product, in terms of three things. First, is payload - so how many people can you hit with any given viral blast. Second, is conversion rate, and third is frequency. This gives you a fundamental idea of how viral a product is.

Hotmail is the canonical example of brilliant viral marketing. Back when Hotmail launched, there were a bunch of mail companies that had been funded and were throwing huge amounts of money at traditional advertising. Back in that time, people couldn’t get free email clients; they had to be tied to their ISP. Hotmail and a couple other companies launched, and their clients were available wherever you went. You could log-in via library internet or school internet, and be able to get access to that. It was a really big value proposition for anyone who wanted to access it. Most of the companies went out there and did big TV campaigns, billboard campaigns, or newspaper campaigns; however, the Hotmail team didn’t have much funding as they did, so they had to scramble around to figure out how to do it. What
they did was add that little link at the bottom of every email that said, ‘Sent from Hotmail. Get your free email here.’

The interesting thing was that it meant that the payload was low: You email one person at a time, you’re not necessarily going to have a big payload. Maybe you send around one of those spam emails, but I’m not sure I’d click on your link. The frequency is high though, because you’re emailing the same people over and over, which means you’re going to hit those people once, twice, three times a day and really bring up the impressions. The conversion rate was also really high because people didn’t like being tied to their ISP email. So Hotmail ended up being extremely viral because it had high frequency and high conversion rates.

Another example is Paypal. Paypal is interesting because there are two sides to it, the buyer and the seller side. The other thing that is interesting is that its mechanism for viral growth is eBay. So you can use a lot of things for virality that may not look necessarily obviously viral. If you said to a seller that you were going to send them money - I can’t think of a higher conversion rate. Frequency was low, and payload was low. But Paypal did this thing where they gave away money when you got your friends to sign up, and that’s how they went viral on the consumer side. They didn’t have to do that for sellers, because if I said ‘I am going to send you money via this,’ you will take that. And even on the consumer side they went viral because if someone says ‘Sign up for this thing and you’ll get ten bucks.’ Why wouldn’t you? So they were able to go viral because their conversion rate was high on the buyer and the seller side, not because their payload and frequency was high. Make sense?

This is a really good way to look at virality if you want to say, ‘Is this product viral?’ Facebook was not viral via email sharing or anything like that. Facebook was purely viral via word of mouth. The interesting thing about Paypal and Hotmail, is to use them, the first person has to send an email to a person who wasn’t on the service. With Facebook, there is no native way to contact people who aren’t on the service. Everyone thinks that Facebook is a viral marketing success, but that’s actually not how it grew. It was word of mouth virality because it was an awesome product you wanted to tell your friends about.

Q: In the first round, it makes sense for there to be a low payload. Will the payload increase in later rounds as the campaign grows and people send more and more e-mails?

A: First and foremost, I think you only send emails to a small number of people. So compared to the massive viral engines that exist today, where you import someone’s entire contact book and send them all an e-mail, or where you post to everyone’s friends on Facebook, the actual payloads are still very small even if it’s everyone that you e-mail on a frequent basis you hit. I’m also thinking per email sent out, how many people are on it. But it’s a fair point that as more people get on Hotmail, they’ll send more emails, and as more people use email, the product grows more and more successfully.

Q: Does a point of conversion matter as well?

A: On Hotmail you click to sign up, but on a billboard you have to remember the URL, go to the website, type it in, find the registration button, click register and sign up. Anything you can do to move friction out of the flow, do it. Going from a billboard ad to an online ad removes huge amounts of friction from the flow.

Q: Are frequency and conversation rate related?

A: Absolutely. If you hit someone with the same email over and over again, or the same banner ad, the same rules apply to every channel. The more times you hit someone with the same Facebook ad, the less
they’ll click. That’s why we have to, like creative exhaustion, rotate creatives on Facebook. Same with banner ads and news feed stories. The fiftieth time you see that IQ story on your news feed, you are not going to want to click on it. The same is true with these emails. So if you send the same email to people over and over again with an invite, you will get a lower conversion rate. ‘The more you hit someone with the same message, the less they convert’ is fundamental across every online marketing channel.

Second way to look at virality, which I think is awesome, is by this guy Ed. Ed runs the growth team at Uber now; he was at the growth team at Facebook. He was a Stanford MBA student, and did a class similar to this where they talked about virality and built viral products. The interesting thing is, if you look at Uber, they’re incredibly focused on drivers. It’s a two-sided market place, so they need drivers. It’s a huge part of their focus as a team, even though they’ve got probably the best viral guy in the world at the company.

So with virality, you get someone to contact import (35:12) that site. Then the question is, how many of those people do you get to send imports? Then, to how many people? Then, how many click? How many sign up? And then how many of those import. So essentially you want people to sign up to your site to import their contacts. You want to then get them to send an invite to all of those contacts - ideally all of those contacts, not just some of them. Then you want a percentage of those to click and sign up. If you multiply all the percentages/numbers in every point in between the steps, this is essentially how you get to the point of ‘What is the K factor?’ For example, let’s says 100 people get an invite per person who imports, then of those, 10% click, and 50% sign up, and of those only 10 to 20% import, you’re going to be at 0.5 - 1.0 K factor, and you’re not going to be viral. A lot of things like Viddy were very good at pumping up stories. They got the factor over 1, which is perfectly doable. But if you’ve got something that doesn’t have high retention on the backend, it doesn’t really matter. You should look at your invite flow and say ‘okay, what is my equivalent to import, how many people per import are invites sent to, how many of those receive clicks, how many of those convert to my site, how many of those then import,’ in order to get an idea of you K factor. The real important thing is still to think about retention, not so much virality, and only do this after you have a large number of people retained on your product per person who signs up.

A couple more things we are going to touch on: SEO, emails, SMS, and push notifications.

In SEO, there are three things you need to think about. First one is keyword research. People do this badly all the time. So I launched this cocktail site I told you about, I spend a year optimizing it to rank for the word cocktail making, but it turns out in the UK, no one searches for cocktail making- about 500 a month; I dominated that search, it was awesome! 400 visitors a month, it was amazing. Everyone searches for cocktail recipes, and in the U.S., everyone searches for drink recipes. So I optimized for the wrong word. You have to do your research first about what you’re going to go after.

Research consists of, what do people search for that’s related to your site, how many people search for it, how many other people are ranking for it, and how valuable is it for you? Supply, demand, and value. So, do your keyword research to figure out which keyword you want to rank for. There are many great tools out there. Honestly the best one is still Google AdWords.

Once you’ve done that, the next most important thing is links. Page ranks is essentially how all SEO is driven, and Google is based on authority. Now there’s a lot of other things in Google’s algorithm now, like, do people search for your website, there’s a lot of stuff about what the distribution of what the anchor text is that’s sent to your site, so that if you abuse it or spam it, they can pop out with spam. White text on a white background five pages below the fold doesn’t work anymore.
But the single most important thing is to get valuable links from high authority websites for you to rank in Google. Then you need to distribute that love inside your site by internally linking effectively. We launched SEO in September 2007; I joined Facebook November 2007. When we launched it, but we were getting no traffic from the pages we had launched, public user profiles. So when I went in and looked at it, the only way you could get into any public user profile was to click on the foot of the page for the about link, then click on the blog articles, then click on one of the authors, and then spider out through their friends to get all their friends.

Turns out that Google was like, ‘They bury these pages, they’re not very valuable. I’m not going to rank them.’ We made one change: We added a directory so that Google could quickly get to every page on the site, and we 100Xed SEO traffic. Very simple change drove a lot of upside by distributing the link love internally.

The last thing is that there’s a whole bunch of table stakes stuff for XML sitemaps, and making sure you have the right headers; it’s all covered really well online for you.

Email is dead for people under 25 in my opinion. Young people don’t use email. They use WhatsApp, SMS, SnapChat, Facebook; they don’t use email. If you’re targeting an older audience, email is still pretty successful. Email still works for distribution, but realistically, email is not great for teenagers - even people at universities. You know how much you use instant messaging apps, and how little you use emails. And you guys are probably on the high scale for email because you’re in Silicon Valley. That being said, on email the things to think about: Email, SMS, and Push Notifications all behave the same way. They all have questions of deliverability, so to finish to finish first, first you have to finish. Your email has to get to someone’s inbox. So if you send a lot of spam, end up with dirty IPs, or send email from shared servers where other people are sending spam from, you are going to end up being put in the spam folder consistently and your email will fail completely. You may end up being blocked and have your email bounce. There’s a lot of stuff around email where you have to look when you receive feedback from the server (42:45) you are sending emails to, 500 series errors (42:49) versus 400 series errors; you have to be respectful how those are handled. If someone gives you a hard bounce, retry once or twice and then stop trying because if you are someone who abuses people’ inboxes, the email companies spam folder you, and it’s very hard to get out. If you get caught in a spamhaus (43:05) link, or anything like that, it’s very hard to get out. It’s really important with email that you are a high class citizen, and that you do good work with email because you want to have deliverability for the long run.

That counts for Push Notifications and SMS, too. With SMS, you can go buy SMS traffic via grey routes (43:24) with people who are having phones strung up attached to a computer and pumping out SMSs. That works for a time, but it always gets shut down. I’ve seen so many companies make these mistakes where they think they’re going to grow by using these kinds of tactics. If you can’t get your email, SMS, or Push Notification delivered, you will never get any success from these. You actually spam your power users and give them notifications they don’t care about, making it really hard for them to opt out. Well, they start blocking you, and you can never push them once they’ve opted out of your Push Notifications. And it’s very hard to prompt them to turn them on once they’ve turned them off.

So number one thing to think about regarding email, SMS, and Push Notifications is, you have to get them to live it. Beyond that, it’s a question of open rate (44:14), click the rate. So what is the compelling subject line you can put there so the people can open your email, and how can you get them to click when they visit?

Everyone focuses towards doing marketing emails that are just spam in my opinion. Newsletters are stupid. Don’t do newsletters because you’ll send the same newsletter to everyone on your site. Someone
who signed up to your site yesterday versus someone who’s been using your product for three years - do they need the same message? No.

The most effective email you can do is notifications. So what are you sending? What should you be notifying people of? This is a great place where we’re in the wrong mindset. As a Facebook user, I don’t want Facebook to email me about every ‘like’ I receive, because I receive a lot of them since I have a lot of Facebook friends. But as a new Facebook user, that first ‘like’ you receive is a magic moment. Turning on notifications throughout all of our channels, increased on our emails, SMS, and Push Notifications, but we only turned it on for low-engaged users who weren’t coming back to the site, so it wouldn’t be spamming for them.

So it was a great experience to think about that. The first thing you need to think about when sending emails, SMS, and Push Notifications is what notifications should we be sending. The second thing you need to be thinking about is how can you create great triggered marketing campaigns. When someone created their first cross-border trade transaction was one of the best email campaigns I was ever part of at eBay in terms of click through rate. It was awesome because it was really timely, and really in context - the right thing to do for the user.

I’d say make sure you have deliverability. Focus on notifications and triggered based emails, SMS, and Push Notifications.

There’s one thing I wanted to finish with, which is my favorite quote by General Patton. It’s so cliche; it’s crazy, but it’s awesome.

“A good plan, violently executed today, is better than a perfect plan tomorrow.”

And one other thing that _____ (46:28) instills in us and Mark still instills across the whole of Facebook is move fast and don’t be afraid to break stuff. If you can run more experiments than the next guy, if you can be hungry for growth, if you can fight and die for every extra user and you stay out late at night to get those extra users to run those experiments to get the data and do it over and over and over again, you will grow faster.

Mark has said he thinks we won because we wanted it more, and I really believe that. We just worked really hard. It’s not like we’re crazy smart, or we’ve all done these things before. We just worked really hard, and we executed fast. I strongly encourage you to do that. Growth is optional.
The transcript hasn't been cleaned up.

Chamath Palihapitiya

The transcript hasn't been cleaned up.

Chamath: I want some intro music from Ignacio. Where is Ignacio? First of all I was playing poker until really late last night so I'm a little rough around the edges. I got my slides to you at midnight. I was at the table doing the slides. I'll just start with a little bit about myself. I'm actually going to give you a somewhat cynical view of this entire space of what it means to be a growth hacker. So don't be offended.

A little bit about me. I think Aaron just touched upon this. Actually, I graduated from EE and in the midst of trying to find my way I ended up working at an investment bank for a year trading derivatives. It was probably the most insipid idiotic use of my time. The biggest thing that I realized was that bankers are smart enough to be greedy but not smart enough to be useful.

So I quit and I applied to all these jobs online and I got one at Winamp accidentally. We actually build something really cool. But the thing that I learned at Winamp, and all of this ties together so I apologize the rambling, but we actually had one of the most important and early consumer platforms available, which was this set of abstractions that we had build to allow people to customize our product. You build skins, you build plugins, and what it actually did was start to refine my understanding of how when you develop things that appeal to specific sets of consumers you start to get this operational leverage in your product.

We were seven people. The things that we did as a seven-person group were pretty impressive back in the day. 100 million users is not as impressive as it is today as it was back then when the internet was a lot smaller but it was still quite an interesting product and really defined a lot of the characteristics of how I actually approached things when I was at Facebook. Winamp was acquired by AOL. At AOL I bumbled along into a bunch of different jobs and ultimately was able to run AIM and ICQ which was the instant messaging product.

The takeaway from that were two things. One, and I'll get to this a little bit later, most people at most companies are really shit. That was manifested in large degree at AOL. I learned all the things not to do, one, and that was probably 90 percent of my learnings to be quite honest with you. Then 10 percent was reinforcing some things that I learned at Winamp which is really about core product value. What it means is create a real connection with someone. I think now we all euphemistically call it the “Aha” moment with the consumer. But also the power of how these communication networks when they develop create real entrenched usage and scale, and how these things can just dramatically accelerate adoption and engagement.

Then lastly I spent a bunch of time at Facebook. The Facebook story is actually really interesting and I'll tell it sort of interleave it as I go through the rest of slides. I spend most of my time investing and playing poker to be quite honest with you. Some of the big ones that I was fairly large stakeholder in Playdom, Yammer which we sold for about two billion dollars in combined value. I'm a fairly large investor in Box, Palantir, and a bunch of very small companies now, some of which where I act as a co-founder, some of which I don't.

A lot of people like to tell every return that they have on their blog. I don't blog or use the Twitter, but the track record here is pretty good. I'm a dad, I have two kids, another one on the way in January. I play a lot of poker and I own the Warriors. That's what I mean, baller for all of you. For all you people don't use your open dictionary, just use this one, okay?

All right, so let's talk about growth. Everyone asks me because this is like the topic du jour, it's what everyone wants to know, it's like what was the secret, it's almost as if we're the NSA and we've developed something and nobody knows, or some secret backroom negotiations between us and government. It's none of that shit. We did three really obvious brain dead things and the reality was we lacked enough self-awareness and ego to frankly just continue to do these very simple things over and over and over again, repetitively, monotonously to a point where every time we used to see things move in one direction or another we would either keep doing them or stop doing them and not second guess ourselves.

I tell people, “You know, look, we actually just looked at a lot of data, we measured a lot of stuff, we tested a lot of stuff, and we tried a lot of stuff.” Now that masks over a lot of more nuanced understanding but at an extremely high level that's really what we did. What's shocking to me is when I see a lot of products out there it's unbelievable to me that people are trying to shroud products in this veneer of complexity that makes themselves seem like so good and so smart and, "I'm this fucking hipster," and, "I'm riding the Muni in San Francisco," and, “Look at my Oolong green tea that I bought organically," and, “Look at these rip tight skinny jeans that I bought.”

It's like you've got to get over yourself. Measure some shit. Try some shit. Test some more shit. Throw the stuff that doesn't work. It's not that complicated. I see app after app after app and I get inundated. When I download and I try them I'm like, “Did you even spend eight seconds using your own product?” It's unbelievable the lack of dogfooding that happens.

So most people when they think about growth they think it's this convoluted thing where you're trying to generate these extra normal behaviors in people. That's not what it's about. What it's about is a very simple elegant understanding of product value and consumer behavior. When you shroud
yourself and all the bullshit veneer, and this is the single biggest problem in the valley today, you will miss the mark.

The problem is we're in this massive long tail where you've had these seminal huge successes occur and now you have all these people who have two choices and they're extremely difficult choices. Choice number one is you do what you think is right, independent of what the external feedback tells you. Choice number two is you do what you read about, and what you get credit for, and what people tell you is interesting. That second-class of things destroys products and it destroys people's ability to build something interesting.

It doesn't matter how good you are at your job if that specific set of values isn't imbedded in what you're building. You will fail and it doesn't matter. Right now that is the one most important thing that if you're going to leave away with this is just don't believe the hype and the bullshit that we're in right now.

How do we do this? We didn't even come up with this framework. A guy worked for me. I'm not going to say who it is because I was literally like nine times I was going to fire this numb skull. Once he came to me and said, “Chamath at eBay we had this framework.” Once he came to me and said, “Chamath at eBay we had this framework.” EBay, this is 2000. I'm like, “EBay, eBay sucks. What can I learn from eBay?” He said, “Well, we had this framework we used at eBay.” We tweaked it a little bit and what I realized was, “Oh my gosh, you know, there's this massive amount of complexity when expressed in simplicity can be extremely useful.”

The tweaks that we made, eBay is a different product and eBay now is a wonderful company doing really, really well. What? Come on, it's doing really well. Was we created a framework in which we applied those three very simple principles of measuring, testing, and trying things. We said, “Okay, the biggest risk that we had is we'd alienate the people that trust us today and use the product.” When you alienate someone what happens is it's actually not palatable generally in top level metrics but there's just this extremely long tail.

Anecdotally you can look at companies, and not to pick on HP as an example, but you look at HP. You're signaling me? 10 minutes? I have 10 minutes left? Jesus Christ, okay. You ask yourself, I know Meg Whitman. She's actually a really great CEO. So what happened? Well, there's no product innovation right now and we have to figure out where their growth comes from. When you trace that thing back it's a decision that was made maybe five or six years ago when it was all about cutting costs and optimizing for structure revenue. So you realize, “Okay, well that's the long tail. That's how long it takes these things to manifest.”

Similarly my biggest fear was we spam our users and we trick them and it will alienate these people. You won't see it today but you'll see in three years from now or four years from now, and it accelerates when you compound that with a competitor who actually builds a better product that doesn't alienate people. The most important thing that we did against our framework was I teased out virality and said you cannot do it. Don't talk about it. Don't touch it. I don't want you to give me any product plans that revolve around this idea of virality. I don't want to hear it.

What I want to hear about is the three most difficult and hard problems that any consumer product has to deal with. How to get people on the front door? How to get them to an “Aha” moment as quickly as possible? And then how do you deliver core product value as often as possible? After all of that is said and done only then can you propose to me how you are going to get people to get more people. That single decision about not even allowing the conversation to revolve around this last thing in my opinion was the most important thing that we did.

When I look again in the landscape, things that scale understand that principle, whether it's explicitly or intuitively, and things that don't and also things that have this amazingly steep ride and then fall off a cliff and there are really visible examples of that today also ignore that principle. It's the discipline to not optimize for the thing that gives you the shortest and most immediate ROI because that is never the sustainable thing that allows you to build something useful.

So when you boil that all up the most important two high-level takeaways that we had and after all of this stuff was we got to eliminate ego. Ego manifests itself every day. I talked about it earlier. It's the ego of basically living a lifestyle and a vision and like a Twitter stream than it is actually like living the life of an entrepreneur building good product and trying to deliver core product value. That takes ego, meaning you have to be comfortable not being rewarded in the short term.

Then the second is to invalidate all the lore. In any given product there's always people who strut around the office like, “You know, I have this gut feeling. It's all about gut feeling.” Most people with gut feeling are fucking morons. They don't know what they're talking about. They just don't. If we lived on gut feeling you can look at what happens when you live on gut feeling. Look at the financial markets, look at how government works, look at how all of these industries that are completely broken. Gut feel is not useful because most people can't predict correctly. We know this.

One of the most important things that we did was just invalidate all of the lore. As much as we didn't do stuff all we did was disprove all of the random anecdotal nonsense that filtered around the company. “Well I think it's this, and people are using it because of that, and I want do this.” It's like, “Where did you pull that out of?” You know where they pulled it out of. Again, you want to do it, to go back and reinforce a sense of ego.

A lot of people don't have a culture within a company that allows these two things to happen. If you can't be extremely
clinical and extremely unemotionally detached from the thing that you're building you will make these massive mistakes and things won't grow because you don't understand what's happened.

It takes a really special type of person to not believe the bullshit, and an even more special person to not conflate luck and skill. You have to be, if you're in this type of job in my opinion, relatively cynical. You can be confident. Fuck, you can be arrogant, it doesn't really matter. But you can't believe your own BS, because when you do you start to compound these massively structural mistakes that again don't expose core product value and then don't allow real engagement and real product value to emerge. You don't listen to consumers because you think it's all about your gut. You don't bother doing any of the traditional straightforward obvious things that would allow you to answer very straightforward obvious questions, and you lose yourself. Most people unfortunately just don't know what they're talking about. I hate this letter. This letter is the dumbest letter in the alphabet. You people are doing more when you focus on this to ruin the internet for the entire human race. Don't talk about this anymore. Just stop. Talk about being in the weeds and not understanding what you're doing. There's no context when you talk about this. None whatsoever. You are spammers, and spamming is pathetic, and it ruins the experience. Don't do it.

We never talked about this once. It never came up once. I didn't have some little guy tickling the ivories on his little Excel spreadsheet, telling me what k values were. Tell me how I'm acquiring people, tell me how we're doing getting them into their "Aha" moment, and tell me core engagement.

Don't give me these low-level abstractions that allow you to validate, get short term results in ROI that don't mean anything. Don't focus on things that destroy long term value. Don't give me stuff that allows you to trick yourself into thinking you know what you're talking about. I'd rather you say you don't know and I'd rather us to figure stuff out together. What I don't want to have happen is a culture where you take these short term things, you start working on it in the absence of context, and you have these meteoric rises and what you have is massive turn fall off and everyone's looking around with their hands in the pockets thinking, "Well, what just happened. I thought I was doing a really great job." You're not doing a really great job. You optimized a variable. Now there maybe somebody on your team that should be doing that, but they should be doing that in the context of something much more important. So you destroy a lot of value when you abstract away that high-level goal to something so ridiculously stupid.

I'm raining on everybody's parade today. Core product value is really allusive and most products don't have any. I actually fundamentally believe that. But I also believe that most products can have some value. So when you put those two things together, again you go back to the discipline of do you really know what your building and why? Do you really understand how to marry things that maybe non intuitive for people, but really are the important things that people need?

I remember when we were launching in Asia. We created a team. The history of Facebook just very quickly. I started Facebook. My team launched Facebook platform, huge success. Then we do this big deal with Microsoft, we raise a huge round. At the tail end of that around it's like, "Oh, we're going to launch a bunch of ad products to validate this valuation, and we're going to do all this stuff." My team goes out and we built three products.

Again, talk about conflating luck and skill. I had all these people, all of our best guys working on this product with me which we called Socialize and [Beak 17:15]. Then we had another thing doing sponsored stories, and then we had another team focused on an online self-service ad product. Most of the resources in mindshare, none of the resources in mindshare. Eight people our best engineers, product managers two people.

Fast-forward to today, all of the revenue, billions of dollars, scaling inordinately, lawsuits. FTC suing us. People telling me on the New York Times I lied about how cross-site scripting worked. I'm not going to lie about cross-site scripting to the New York Times. New York Times, okay, guy, I mean really? That's I'm going to lie to you guys about? So stupid. It just goes to show you at that point we were seeing this what looked like monotonically negative growth and we're like, "Oh my god, what's happening?" That's like 25, 30 million people. We came up with this idea, "Okay, we're going to focus on growth." We went created a team.

As part of that we said, "We're going to internationalize, we're going to launch, and we're just going to go everywhere all over the world." Again, we dogfooled our own product and we used Facebook platform to create this crowdsourcing translation which was really successful. But long story short we get to Japan, Korea, and it's a constant refrain. I go around the valley. I talk to some folks who have done this before, the guys at Yahoo, the guys at Google, the guys at eBay, just to get a sense of how did you think about it. All these people had the same answer. "Well, we take these MBA's who really want to live on an expat package and we send them out there." You're just like, "Okay, we're going to focus on growth." We went created a team.

First thing we did was we hired only native people in those native markets. I had these guys trying to be, "I have an MBA from Harvard. You know, I really want to spend time." I'm like, "Get the fuck out of my office. Go to ba.com buy a ticket, get out of my face. You're not going to travel on the company dime." But you hire all these people natively. We had this amazing guy that we hired in Japan. [Inaudible 19:11] said, "You know, it's really important to actually have specific elements of the profile be different because in Japan the culture dynamic is different." We said, "Well, what does that mean to
you?” He said, “Well maybe it means you know, putting your blood type on the profile.” You’re like, “That makes no sense.”

But it makes no sense to us. But it makes sense to him. So we developed this flexibility where were like, “Okay, well, maybe the connections we’re trying to make in that market are a little too allusive.” This product that worked here just didn’t have any context here. You would say, “Well, Facebook has clear product value,” but in Japan at the time it didn’t.

So having the courage to reset and redefine what it means in any given market again takes a lot of courage. We did it, and now it’s massively scaling. Then all of the sudden the light bulb goes off and you’re like, “Oh my gosh, it’s like we don’t know what we don’t know.” In every single market people react differently, they behave differently, they speak different languages. Guess what, Spanish is not Spanish. Maybe they know that. Five minutes. I got it. I’m on it.

The point is even when you think something works it probably doesn’t work for everyone, and finding a framework that allows you to actually restructure and reorganize the things that you’re doing to expose value in different ways to different people in different situations is an extremely important thing. Again, it goes back to how you live in the world where you’re willing to redefine and reset the things that are independent of what the short term feedback loop is telling you. It’s a very difficult proposition.

We knew we were going to beat MySpace when we had 45 million users. They had 115 million. We just knew. We used to just like, “That was not what we celebrated.” We celebrated at 45 million. It was like a high five. We’re like, “All right, let’s get back to work.” The reason was because we had started to do enough things right where we could just see now we understood what we’re doing. After all the testing, all the iterating, all of this stuff, you know what the single biggest thing we realized? Get any individual to seven friends in 10 days. That was it. You want a keystone? That was our keystone. There’s not much more complexity than that.

There’s an entire team now, hundreds of people that have helped ramp this product to a billion users based on that one simple rule. So if you were looking for a lot of complexity I couldn’t give it to you. But we were able to reframe the entire experience around that one simple premise, a very simple elegant statement of what it was to both capture core product value, to define what it meant to be able to onboard into a product that allowed you to communicate, to get into a network, to find density, and then to basically iterate around that.

Then what we did at that company was we talked about nothing else. Every Q&A, every all hands nothing was spoken about other than this. Monetization didn’t really come up. Platform came up but again in a secondary or tertiary context. But it was the single sole focus. But because we had defined it in this very elegant way that expressed it as a function of product value it was something that everyone could intrinsically wrap their arms around.

Again another reason why focusing on abstracting growth down to these low-level things is not right. The person that runs growth on any team has to be strategic and capable enough to engage on the core strategy of the product. Because then you allow yourself to up level the conversation and have it become the most important framework in which you organized an entire company, how equity is given, how expenses are generated, how things are focused on, who gets hired, who gets fired.

But when you understand core product value and then you could basically pivot around it and create these loops that expose that over and over and over again this becomes the most important and obvious thing to do. In the few companies that I help now with this context, when they do that and when it’s the CEO or when it’s the cofounder who are living in that world and living in this context they’ve consistently systematically every single one has been successful, every single one of them.

I’ve also meet a lot of people who ask all these questions. They’re like, “Hey, I’d love to really …” I meet them and they’re kind of d-bags. They’re like way too quantish and way too passive aggressive and a little agro. It reminds me of something that’s actually again a lot more important than the short term ability for people to focus on short term results, which is in this battle between culture and capability culture wins every time. What you value is really what you achieve. So when you see these companies again today that had these massive apexes like rocket ship growth and now some are public and now are going through just unbelievable wrenching, gut-wrenching change, negative spiraling turn downwards, what you realize is that’s not a growth problem, although it seems numerically a growth problem. What that is is a product value and culture problem.

When you’re focused on again KKKKKK. Well, not KKK but you get the point. Another reason why K sucks, right? I mean use it three times you’re racist, like Jesus. But the point is that’s the type of stop that always comes back and it gets you. To short term optimization it never works. You have to work backwards from what is the thing that people are here to do, what is the “Aha” moment that they want? Why cannot they give that to them as fast as possible? Measuring that in days is unrealistic. Measuring it in hours is unrealistic. Measuring it in minutes is necessary but not sufficient. But like, “How do you get that to seconds?” How do you get that to hundreds of milliseconds?” That’s how you win.

If you even can’t understand what the thing does optimizing all of this stuff over time I think it just creates these really bad crappy companies that all they do is just spam and destroy the internet for everybody else.
To this end this is really what I want to leave you with, which is you probably expected a bunch of formulas and stuff. You have to really understand this. Go back to that first slide, detached, egoless, focused on what's important, not living the hype, not trying to follow some lifestyle, but doing what you think is important, focusing on core value, ruthlessly prioritizing, and getting people who believe in these things. These are the values that we originally wrote in terms of how we recruit.

Again, it may seem like a crazy thing but for all the CEOs here all of you people should be writing this down. This is how you should recruit. It worked for us. I give this to every single company I invest in. It works for most of them. These are things that in my mind are so obvious, blindingly glaringly obvious. People make mistakes around these things all the time.

A very high IQ, self-explanatory, a strong sense of purpose are people that will not buckle under short term pressure, a relentless focus on success just so competitive that you will do whatever it takes to win, aggressive and competitive people who always respect the person but constantly challenge the idea, a high quality bar that borders on perfectionism, just nothing is ever good, you win, something improves by 10 basis points, now it needs to improve by 20 basis points, now it needs to improve by a percent. That living in that world is just a really great dynamic. People who are comfortable taking something that works and saying it doesn't work here unless just reset start again. It just takes a lot of courage.

When that guy came to me Japan and said, “Hey, I think we need to introduce a blood type,” or our team in Korea came up with a bunch of different ideas, or our team in India came up with this really clever way of engaging with Facebook over a phone, you have to be willing to take yourself out of your own biases and find people who are willing to come up with things that again are unconventional, another reason why you can't live in the bubble that is the valley.

There are too many problems and too many products that are focused on solving the needs of the one percent of the one percent that live here. The reality is there are seven billion people in the world. Most have never even used the PC. Most are coming straight into a world of phones. They're living in entirely different context, they have completely different socio-economic paradigms, and you have to be able to be sensitive enough to empathize with where they're coming from.

High integrity I think speaks for itself. It's really easy to focus on short term results. I just don't think there's enough of the long term thinking, being able to take some arrows along the way in the short term because you're trying to build something for the long term, and having a culture and set of values where that's rewarded.

A perfect example of this is LinkedIn. Reid Hoffman is an extremely good friend of mine. That is an unbelievable company, built over 10 years 11 years of just working, working, working. Friendster comes out of nowhere. We didn't panic. MySpace comes out of nowhere. We didn't panic. Facebook came out of nowhere. We didn't panic. Just built a great product, we just kept working and working, did not take the short easy way out. You have to respect that because when those guys win they win and they win in scale and they win for a long time.

Surround yourself with good people seems obvious but I see a lot of people who are just not comfortable recruiting people that are better than them. The best thing that I did was recruit four people to my team. My original growth team was James Wang, who was the engineering lead on Facebook platform, Naomi Gleit who was the most tenured employee at Facebook and one of our best product managers, Alex Schultz who is this unbelievable crazy growth guy, and Javier Olivan who ran International for me, and Blake Ross who's the product manager and also founded FireFox. All of these guys were dramatically better than I was. All I did was enable a framework and create discipline, and they all thrived. The great thing when I left was didn't miss a beat. That's really, really a good sign.

Then the last thing is cares about building real value over perception. I like Oolong green tea as much as the next guy. It doesn't make anything happen. I like skinny jeans too, more on girls than guys but whatever. It doesn't mean anything. Right now we're living in a world where you can get distracted by things that don't matter and the superficiality of how you think things should be done versus the things that need to get done. That takes a lot of courage.

I think my time is up. Thank you. I'm happy to take questions.

Speaker 2: We have a few minutes for questions. We're trying our new approach here of Tweeting questions. One of the first ones we have is what was it about Instagram that made it worth one billion dollars to Facebook?

Chamath: I'll give you my opinion. Again, as a person that's not involved but was there. Again, when you talk about core product value at Facebook one of the key things that it got right or it understood was the connectivity around, you being able to associate your friends in social situations. The simplest manifestation of that was in photos. When we launched our photo feature it literally was just explosive growth. Here it is where in many ways it is the atomic unit of how Facebook gets organized. As much as people are in atomic units it's almost even more important in the sense that it's even more element...

All that is great. Facebook is generating billions and billions of photos a day. All of a sudden along comes this company who does the equivalent thing in a slightly different paradigm, a looser privacy model, but all on mobile, and in a context where people enjoy consuming the content as much as they enjoy consuming the photos on Facebook.
From my perspective is both defensive and offensive. From the defensive perspective it locks in that core tenant of behavior and product value that it's essential and the element of building blocks on which everything else on Facebook is built on top of. Offensively it gets you into a place where now you have this flanker strategy where you have your main product getting more and more usage and getting more and more entrenched and creating this fertile ground all over the world on phones, and you have this unbelievable product specific application that does the same thing. Now you both can go and thrive.

But it was a combination in my opinion of the offensive capability of what it gave Facebook as a tool in its arsenal and defensively protecting its core element that if all of the sudden that behavior seeded someplace else and photo activities started to be generated and stayed in another environment I think it's very problematic for Facebook. Or would've been, but I don't think is it.

Speaker 2: All right, actually anyone else with a new tweet question? Everyone's busy quoting you Chamath so lots of good quotes today.

Chamath: I don't use the Twitter but I'll make sure …

Speaker 2: We have time for one more if someone wants to … We're waiting for an update here.

Audience: [Inaudible 33:14]

Chamath: Well it actually … No, that's that's exactly right. So it started with invalidating lore. We had all of these anecdotes about how people thought the side worked, and part of me, it's just the cynical part of me which is like I just like proving people wrong, especially when I know they don't know what they're talking about. For the first six months it was like we were just going to instrument something and just prove people wrong because the best thing is what they do is a shut up. Then actually the next time they speak they try to know what they're talking about, which generally culturally is just a very good thing.

In that process we're like, "Oh my gosh, there's these weird things that are happening." You rinse and repeat through all these cohorts and you're like, "Why did these people get into a certain space and are now fully engaged and ramped-up and have these vibrant networks, and these people didn't?" We just tried enough things where we were able to back into that axiom being the definitive thing that defined how things work for us. But for you it's going to be different. But it started.

I mean if I have to give you a framework it would probably be you've got to start with a broad cross-section of engaged users and when you work backwards from each of those and you are smart enough and clever enough to really figure out the different pathways in which they got to that place you can probably tease out what those simple things are, and then hopefully, I think most products are structured this way, you can then path people, more and more of those people into those same clip flows that allow them to get to that state.

But again it starts with looking at an engaged user, not just thinking about how many emails can I send and how do I trick everyone to click on a select dollar and not unselected the selected dollar. No offence, we did that too, but I mean we did that after we figured that shit out.

Speaker 2: Okay we'll take one last question.

Audience: [Inaudible 35:14]

Chamath: Look, the best thing that happened is I'm not American, I'm Canadian. But even I'm not even really Canadian. Well, I am Canadian but I was born in Sri Lanka. I like, I love America and everything that is given to me, but I'm pretty cynical about the imperialistic nonsense of a lot of Americans and how Americans treat their place in the world to be quite honest with you.

My mental worldview I think was one where that combined with the feedback that I got from all these big companies like, "Well, you know, I have this really great, you know, MBA, JD guy that, you know," and it's just like, he was like, "Well, how is this whitey going to fucking figure out Korea?" No offense but Korea is going to figure out Korea before. You know what I'm saying? This guy is, "Hey, guys, we're coming here to take over." It's like, "Really? That's crazy. That's just absolutely crazy." I was just like, "Let's just hire some really smart awesome young Korean lady," and she's kicking ass. Then it's like, "Let's do the same thing in Japan, let's do the same thing in Brazil, let's do the same thing …" It's not complicated. That's not so hard, and then just listen to them and let them try.

We had culture fortunately of experimenting and trying at ton of stuff and we celebrated failure more than we really celebrated success. We would celebrate these massive landmark milestones, 100 million, 250 million, 500 million, and even one billion, but one billion was like how do I call [inaudible 36:44].

The growth team had a dinner here at Madera. We all got together, the original team and had a quiet celebration. We are more interested in just trying stuff and learning. Then it was just a matter of having different people willing to try different stuff in their own markets and then us being able to give them a framework where we said, "Look, 99 percent of things can't change but you have this one percent flexibility and if you need more justify it." That was a good interplay. But it's was just part of my worldview. America's great but there's a lot of great people and a lot of great countries too.

Audience: [Inaudible 37:22]

Chamath: I think those frameworks, the great thing is we had to build all of our stuff ourselves. You don't have to do that anymore. For you guys like even something as simple
as like ... Our ability to AB test was just a convoluted mess. For you it's like [inaudible 37:48]. You can't even you do it yourself just go use optimizing. Or now there's like 19,000 gazillion Hadoop companies and Hive companies out there. There was none when we were doing this stuff. We were rolling our own stuff constantly. We just had a lot of delays that you don't have to suffer from.

That abstraction allows you to free up resources that you can allocate in different places. But again, it comes back to the discipline of like is the senior manager of the company fixated on a goal and then giving flexibility to someone who has the political capital and the patience and the resolve to get there, and then the ability to question and understand really what's happening in their product?

The other thing is a lot of this credit goes to Zuck because he managed the board and all of the people around so that it's like we're doing this. Do you know what I'm saying? Sometimes it's like ... We had a very functional useful board. A lot of times boards can maybe distract CEOs as well. “Well focus on this and focus on that.” Then they come down they shout for the mountain tops, “Hey, we're going to do this.” Then when it gets filtered down to an individual person what they hear is, “KKK KKK.” I think that's where you lose your stride.

Speaker 2: All right, thank you guys. Thank you Chamath. Thank you. A round of applause.
Lecture 6: Does Slow Growth Equal Slow Death?


Joel Spolsky

I always thought that expanding my business at a steady pace was a smart move. Now I worry that it could potentially kill us.

One of our best programmers just came into my office. He's working on a new price list that we’re putting into place, and he's on a tight deadline. I need to be able to announce the new rates in three weeks, at the first public demonstration of the application FogBugz 7.0. Implementing the new prices on our website is going to take too long, he told me. We aren't going to have it ready in time for my announcement. "What? You’ve got three weeks! Three weeks to change some prices? I could change those prices by the time I go home today," I lied. Changing the prices for new customers is easy, he explained. The hard part is building a slick new area on our website where existing customers can go to convert to the new pricing. "What if we don't have a slick website?" I asked. "Anybody who wanted to convert would have to call in, and we’d have to do it manually," he said. "Could you get that part done on time? With the manual conversion?" He said he could. Which means we’re going to do a half-baked job of implementing these new prices. We have to, this time, because we’re committed to a date. Normally, we don’t do things this way at Fog Creek. Speed to market usually involves a direct tradeoff with quality. If you need high-quality code, it takes time, and we’ve always taken the time to do things right. No deadlines here! Among other things, taking the time to do things right has probably slowed us down. A lot.

We’re doing great, thanks for asking, but after nine years, I’m beginning to accept the evidence that whatever it is we’re doing results in reasonable, steady growth, not spectacular, explosive growth. We’ve gone from two to 25 employees in about a decade by growing, on average, 54 percent a year.

I have always believed that there is a natural, organic rate at which a business should grow, and that if we expanded too fast, the wheels would come flying off. (We’d have customers calling us up left and right to do things manually that should have been handled online, for example.)

And I have to say, I’ve been happy with our growth rate. I was resigned to never being on the Inc. 500, because growth for the sake of growth leads to all kinds of, well, growing pains. For the longest time, I smugly thought: We’re profitable, our sales are rising, we make terrific products, and our customers love us. So what do we have to worry about?

Then I came across a quote from Geoffrey Moore, who is best known for his best-selling book Crossing the Chasm, which is about how businesses cross over from their initial niche markets to dominate larger markets. In another book, called Inside the Tornado, Moore writes about the great battle between Oracle and Ingres in the early 1980s. The winner of that battle is well known: Oracle now has a market cap of more than $100 billion, and I’ll bet you’ve never heard of Ingres.

"What set Oracle apart from Ingres," Moore writes, "was that [CEO] Larry Ellison drove for 100 percent growth while Ingres ‘accepted’ 50 percent growth." Executives at Ingres meant well. According to Moore, they felt that the company "simply cannot grow any faster than 50 percent and still adequately serve our customers. No one can. Look at Oracle. They are promising anything and everything and shipping little or nothing. Everybody knows it. Their customers hate them. They are going to hit the wall."

Of course, Oracle overcame those concerns and eclipsed its rival. And this got me worried. Were we Ingres? I had to wonder. We do have a large competitor in our market that appears to be growing a lot faster than we are. The company is closing big deals with big, enterprise customers. And the wheels are falling off the donkey cart over there as the company stretches to fulfill its obligations. Meanwhile, our product is miles better, and we’re a well-run company, but it doesn’t seem to matter. Why?

Moore explains that "for pragmatist customers, the first freedom in a rapidly shifting market is order and security. That can only come from rallying around a clear market leader. Once the apparent leader-to-be emerges, pragmatists will support that company, virtually regardless of how arrogant, unresponsive, or overpriced it is."

Uh-oh. Are we actually losing our market leadership position because we’re careful? It’s entirely possible. Think of it this way: If you’re growing at 50 percent a year, and your competitor is growing at 100 percent a year, it takes only eight years before your competitor is 10 times bigger than you. And when it’s 10 times bigger than you, it can buy 10 times as much advertising and do 10 times as many projects and have meetings with 10 times as many customers. And you begin to disappear.

As you may recall, there were lots of cute little word-processing software companies in the late 1980s. Remember WordPerfect? WordStar? Ami Pro? That business has been completely dead since Microsoft Word for Windows emerged as the one to beat in the early 1990s. I’ll bet the Ami Pro product managers were sitting around feeling pleased with their solid annual-growth numbers, just as Microsoft was growing faster and becoming the de facto standard.

Expanding your business at faster than its natural rate is a risky thing to do, of course. You have to hire quickly, which
Lecture 6: Growth

reduces the profit margin. You may have to borrow money or take on investors. You have to rely on outside partners more. You have to trust your employees to do things that you used to be able to do yourself, and nobody does them better than you.

So what's the proper amount of risk? A lot of people would say, "Gosh, zero?" OK, that sounds safe and reasonable. You worked hard to build your business; you're counting on it for your retirement and your kids' college and whatever. But if you're not taking any risks, you're pretty much guaranteed to fail. Somewhere, there's someone out there who is taking more risks than you, and that person's business is growing faster than yours, and that person's business may one day come to dominate your industry while yours withers away.

Anyway, you didn't become an entrepreneur to be safe and reasonable. Here at Fog Creek, I feel like I can certainly afford to take more risks in pursuit of a higher rate of growth. I have the beginning of a plan for how to proceed.

Step One, I think, is to pluck off our biggest competitors. We're pretty certain that we've already built a great product that meets our customers' needs -- but there are still too many cases where we find out that, for some reason, someone went with the other guy. So that's the development team's mission for 2010: to eliminate any possible reason that customers might buy our competitors' junk, just because there is some dinky little feature that they told themselves they absolutely couldn't live without. I don't think this is going to be very hard, frankly. Developing great software is something I'm pretty sure we're good at.

Step Two is something I'm not particularly good at. Not in the least bit. We have to build up our sales force. The bottom line is, we just don't do enough selling. I've been working on the assumption that a product naturally creates demand for itself and the sales team just helps fulfill that demand. But I've realized that I have things backward. I've come to understand that a sales team drives demand. My problem is that I've never been able to figure out how to hire good salespeople. For a guy who wrote a book on how to hire great programmers, it's mortifying how incompetent I've been at enlarging the sales team, which, right now, consists of one terrific account executive and a dog. (I'm just kidding. There's no dog.)

We don't want to win with lousy products or high-pressure sales tactics. We have no intention of giving up our commitment to good customer service or high-quality code. But we do have to work closer to the limits of our abilities, we have to invest more of our profits in hiring more salespeople and software developers, and we have to focus relentlessly on winning more enterprise sales. We have to do that, because otherwise, we're going to end up being the company you've never heard of.

Joel Spolsky is the co-founder and CEO of Fog Creek Software and the host of the popular blog Joel on Software. For an archive of his columns, go to www.inc.com/keyword/spolsky.
Lecture 6: A Recipe for Growth: Adding Layers to the Cake

http://jeff.a16z.com/2012/01/18/a-recipe-for-growth-adding-layers-to-the-cake/

Jeff Jordan

Businesses don’t grow themselves. One of the most important jobs of a CEO is to aggressively define and pursue a growth agenda for his or her business. Why is this important? Growth typically improves a company’s competitive position and provides increased scale and leverage, and investors clearly value growth.

The pursuit of growth continues to be important regardless of the lifecycle of the company. Obviously it’s critical early in a company’s life… or it won’t be a company for long. But it continues to be important as a company develops. Virtually all businesses, even hyper-growth ones, inevitably experience slower growth as they get larger, with their growth rates falling relentlessly back down to Earth over time. I call this effect “gravity” and it will weigh down even the most promising of companies—unless a CEO can find a way to accelerate growth and positively change the long-term growth trajectory of the business.

The first real operating job I had was managing eBay’s U.S. business in mid-2000, which included the ebay.com website. Virtually all the revenue—and more than all of the profits—of the eBay company came from the U.S. unit at the time, and despite the bursting of the bubble, EBAY was still trading at highly robust multiples. So you can imagine the terror I felt when the U.S. segment failed to deliver month-over-month growth for the first time ever in my first month on the job. The heavy weight of sky-high growth expectations was showing the first signs of a potential collision with the brutal effects of gravity.

It was clear we needed to quickly define a growth agenda that had the scale to fight gravity’s impact. We quickly narrowed the options down to a few: spend more marketing or spend it more efficiently, innovate the product, or buy a company to help us grow.

Marketing had some leverage, but it was limited. eBay was already one of the biggest marketers on the Internet and efforts to optimize spend were already underway. M&A, on the other hand, felt both desperate and was controlled in a separate part of the organization. So we quickly turned our attention to focusing on product innovation.

One of the first places we looked for growth was in buying formats. ebay.com at the time enabled the community to buy and sell solely through online auctions. Many in the community thought this was the magic of the site, and it clearly helped propel the company to a very strong start. But auctions intimidated many prospective users who expressed preference for the ease and simplicity of fixed price formats. Interestingly, our research suggested that our online auction users were biased towards men, who relished the competitive aspect of the auction. So the first major innovation we pursued was to implement the (revolutionary!) concept of offering items for a fixed price on ebay.com, which we termed “buy-it-now”.

Buy-it-now was surprisingly controversial to many in both the eBay community and in eBay headquarters. But we swallowed hard, took the risk and launched the feature… and it paid off big: Buy-it-now complemented auctions well, brought new users and new listings to the site, and became a very important driver of growth for many years. These days, the buy-it-now format represents over $40 billion of annual Gross Merchandise Volume for eBay, 62% of their total.

With an initial success, we doubled down on innovation to drive growth. We introduced stores on eBay, which dramatically increased the amount of product offered for sale on the platform. We expanded the menu of optional features that sellers could purchase to better highlight their listings on the site. We improved the post-transaction experience on ebay.com by significantly improving the “checkout” flow, including the eventual seamless integration of PayPal on the eBay site. Each of these innovations supported the growth of the business and helped to keep that gravity at bay.

I came to call this process of layering in new innovations on top of the core business “adding layers to the cake”. Much of the natural effort in the organization is spent on chasing optimization of the core business. This makes sense, as small improvements in a big business can have a meaningful impact. But there is huge potential leverage to adding layers of new, complementary businesses on top of the core (aka “cake”). In the ebay.com case, buy-it-now, stores, features, checkout and PayPal integration were all new initiatives that layered on top of the core business but added something new to it.

The eBay company in its first decade is a good illustration of the impact of “layers on the cake”. eBay U.S. was the company’s original business, and my team worked tirelessly to optimize it and add layers on top of it. And at the company level, the eBay Inc. management team also looked to add layers. Our first was international expansion, which started in earnest in the early 2000’s. We followed with payments, facilitated by our acquisition of PayPal (and worth noting here that PayPal’s early growth was primarily as the payment functionality on the eBay marketplace). Here’s what the result looked like at the company level:

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Lecture 6: Growth

eBay U.S. clearly was a fantastic business in and of itself, and it demonstrated strong, sustained growth. At the same time, the international and payments layers grew from virtually nothing in 2000 to around 60% of the company’s revenue by 2005. As a result, the overall company grew dramatically faster than its original core business and successfully fought off the impact of gravity for a decade. And the market rewarded the company handsomely for this growth. Here’s EBAY’s stock price during the period:

Source: eBay SEC filings

After eBay, I continued to deploy the layers-on-the-cake approach at the other Internet companies that I’ve managed. At PayPal, the key layers we implemented there were international expansion, improving PayPal’s offerings for merchants who wanted to sell outside of the eBay platform (called ‘Merchant Services’), and starting to offer credit on top of our payments business. We even trialed a text-based mobile payments product in 2006, although the market wasn’t quite ready for it at that time (I’m convinced the product’s developers and I were the only people who ever used it).

During my time at OpenTable, the key layers we introduced included building a robust set of mobile applications that expanded diner use cases, expanding internationally (again), introducing a new “Connect” product that meaningfully increased the addressable market of restaurants, and developing yield generation products that helped restaurants attract additional diners. These initiatives helped OpenTable overcome gravity. For example, year-over-year revenue growth rates accelerated from 23% in 2009 to 44% in 2010.

Two other illustrations of the success of this layering approach are provided by two of the most successful growth companies of the past decade: Apple and Amazon. Steve Jobs and the Apple team relentlessly added new layers at Apple that sat on top of their original core business of computers, including the iPod, iTunes, the iPhone and the iPad. And Amazon in recent years has innovated incredibly skillfully beyond their core physical merchandise business, adding layers such as Prime, digital goods, Amazon Web Services and the Kindle and now Fire digital devices. These very large companies demonstrated explosive growth pretty much entirely through brilliant innovation.

Innovation clearly is THE success model on the Internet. It explains how Google emerged as the dominant player in search, despite being relatively late to market and competing with established companies like Yahoo and Microsoft. It explains how PayPal buried the other online payment sites that started around the same time, including billpoint.com and accept.com, despite these companies having preferred access to the massive eBay and Amazon platforms, respectively. And it explains how Facebook has come to dominate social networking, even though it was very late to market relative to Friendster and MySpace.

These winning Net companies are incredibly strong at product innovation. They invest in it, they create cultures that support it, they prize it and they reward it. The companies above that failed to capitalize on their early success arguably did not. The best innovations improve and compliment the core business of a company, taking advantage of and enhancing its most valuable assets. Diversification outside of the core business is a much more challenging strategy. The further a company strays from its core in its innovation, the longer the odds of success.

I’m a huge believer in the potential for innovation to drive results for all companies, but particularly for technology companies. Core to the CEO’s job is to rise above the day-to-day requirements to keep his or her vision far out on the horizon, proactively delivering new innovations today that have the impact to materially boost the long-term growth of the business in the future.
Lecture 6: Mark Zuckerberg on Facebook’s Early Days: Go Hard or Go Home

http://allthingsd.com/20121020/mark-zuckerberg-on-facebooks-early-days-go-hard-or-go-home/

Mark Zuckerberg

Like any young start-up, the early days of Facebook were thin and scrappy. Its very first server back in 2004 cost $85 to rent. They didn’t spend more than they had in the bank. They were small, tight and still had everything to prove.

To do that, CEO Mark Zuckerberg said, the company needed to test its mettle against its existing competitors. And back then, those weren’t MySpace or Friendster, but the existing social networks inside U.S. universities.

“We first went to schools that were hardest to succeed at,” Zuckerberg said on Saturday morning, kicking off the Y Combinator Startup School event in Palo Alto, California. “If we had a product that was better than others, it would be worth investing in.”

Zuckerberg spoke to a packed house in the Stanford Memorial Hall auditorium, with an audience mostly composed of twentysomethings, the veritable next wave of young Silicon Valley entrepreneurs. The conference is geared toward the young and idealistic, those who may build the Facebooks or Twitters of tomorrow. Hence, Zuckerberg focused on the challenges of turning a rough-and-tumble outfit into the 1-billion-user-strong social giant it is today.

So if you’ll hearken back to 2004, Facebook’s first days were limited to college students alone, those who had verified university email addresses. It was a play for an early conception of true online identity; unlike other existing networks, you were supposed to be yourself on Facebook.

After first growing Facebook inside of Harvard’s network, then, the plan was essentially to go hard or go home — to launch the network at universities like Columbia, Stanford and Yale. These were the schools, Zuckerberg said, that had the most integrated social networks campus-wide. If Facebook caught on here, it’d be safer to assume that scaling to less-integrated schools would be a downhill battle.

That’s exactly what happened. Facebook spread from school to school, moving slowly to cope with the early scaling issues that popular services often face (Twitter and the Fail Whale, anyone?).

Much of the other advice Zuckerberg offered to the young crowd was the usual platitudes — listen to your users, stay simple, be reliable.

But his most important point was clear: Punch above your weight class. If your product is better than anything out there, the users will let you know it.

Lecture 6: The Secret Behind Pinterest’s Growth Was Marketing, Not Engineering, Says CEO Ben Silbermann

http://allthingsd.com/20121020/the-secret-behind-pinterests-growth-was-marketing-not-engineering-says-ceo-ben-silbermann/

Ben Silbermann

Pinterest, which CEO Ben Silbermann describes as a tool that helps people find inspiration, is now the third-largest source of referral traffic on the Internet. But growth wasn’t easy for the company, Silbermann told a rapt audience at Y Combinator’s Startup School at Stanford University on Saturday.

The way Pinterest grew had little to do with Silicon Valley wisdom. It was about marketing — mostly grassroots marketing — not better algorithms.

In 2010, three months after Pinterest launched, the site had only 3,000 users. But some of them were active users, and those people loved the site — and both of those categories included Silbermann himself. “Instead of changing the product, I thought maybe I could just find people like me,” he said.

So Pinterest started to have meet-ups at local boutiques, and to take fun pictures of people who attended them, and to engage with bloggers to do invitation campaigns like “Pin It Forward,” where bloggers got more invites to the site by spreading the world.

Silbermann said he realized the strategy might just be working when he heard people at a meetup having real conversations with each other about their creative projects, rather than the BS that might come from a superficial relationship on Twitter or Facebook.

“A lot of people in Silicon Valley didn’t get, and I don’t know if they still get, Pinterest,” Silbermann said. “The fact that it made sense to someone is what really mattered to me.”

In its earlier days, most investors weren’t willing to buy into Pinterest and its non-technical founders. (These days, that’s no longer the case.) “There are lots of ways for investors to say no to you, and I’m pretty sure I’ve heard every single one,” Silbermann said.

Investors did want to offer all sorts of feedback about what Pinterest should change. A few years ago, VCs wanted things to be text-oriented and real-time, while Pinterest is visual and more timeless. Silbermann told the founders and would-be founders at Startup School that they shouldn’t take VC advice and buy into Silicon Valley groupthink, an argument he has made before.

“Fundamentally, the future is unwritten. If they knew, they would be done,” Silbermann said. Still, Silbermann added that he himself thought for a while that the secret to
Pinterest’s growth woes would be finding some undiscovered Stanford grad student to build a better algorithm. But ultimately, Pinterest didn’t need better engineering, said Silbermann. It needed better distribution. And so if there’s any broadly applicable lesson from Pinterest’s success, he said, it’s that there are many ways to succeed.

Lecture 6: Startup = growth

http://www.paulgraham.com/growth.html

Paul Graham

A startup is a company designed to grow fast. Being newly founded does not in itself make a company a startup. Nor is it necessary for a startup to work on technology, or take venture funding, or have some sort of “exit.” The only essential thing is growth. Everything else we associate with startups follows from growth.

If you want to start one it’s important to understand that. Startups are so hard that you can't be pointed off to the side and hope to succeed. You have to know that growth is what you're after. The good news is, if you get growth, everything else tends to fall into place. Which means you can use growth like a compass to make almost every decision you face.

Redwoods

Let's start with a distinction that should be obvious but is often overlooked: not every newly founded company is a startup. Millions of companies are started every year in the US. Only a tiny fraction are startups. Most are service businesses—restaurants, barbershops, plumbers, and so on. These are not startups, except in a few unusual cases. A barbershop isn't designed to grow fast. Whereas a search engine, for example, is.

When I say startups are designed to grow fast, I mean it in two senses. Partly I mean designed in the sense of intended, because most startups fail. But I also mean startups are different by nature, in the same way a redwood seedling has a different destiny from a bean sprout.

That difference is why there's a distinct word, "startup," for companies designed to grow fast. If all companies were essentially similar, but some through luck or the efforts of their founders ended up growing very fast, we wouldn't need a separate word. We could just talk about super-successful companies and less successful ones. But in fact startups do have a different sort of DNA from other businesses. Google is not just a barbershop whose founders were unusually lucky and hard-working. Google was different from the beginning.

To grow rapidly, you need to make something you can sell to a big market. That's the difference between Google and a barbershop. A barbershop doesn't scale.

For a company to grow really big, it must (a) make something lots of people want, and (b) reach and serve all those people. Barbershops are doing fine in the (a) department. Almost everyone needs their hair cut. The problem for a barbershop, as for any retail establishment, is (b). A barbershop serves customers in person, and few will
travel far for a haircut. And even if they did the barbershop couldn't accommodate them. [1]

Writing software is a great way to solve (b), but you can still end up constrained in (a). If you write software to teach Tibetan to Hungarian speakers, you'll be able to reach most of the people who want it, but there won't be many of them. If you make software to teach English to Chinese speakers, however, you're in startup territory.

Most businesses are tightly constrained in (a) or (b). The distinctive feature of successful startups is that they're not.

Ideas

It might seem that it would always be better to start a startup than an ordinary business. If you're going to start a company, why not start the type with the most potential? The catch is that this is a (fairly) efficient market. If you write software to teach Tibetan to Hungarians, you won't have much competition. If you write software to teach English to Chinese speakers, you'll face ferocious competition, precisely because that's such a larger prize. [2]

The constraints that limit ordinary companies also protect them. That's the tradeoff. If you start a barbershop, you only have to compete with other local barbers. If you start a search engine you have to compete with the whole world.

The most important thing that the constraints on a normal business protect it from is not competition, however, but the difficulty of coming up with new ideas. If you open a bar in a particular neighborhood, as well as limiting your potential and protecting you from competitors, that geographic constraint also helps define your company. Bar + neighborhood is a sufficient idea for a small business. Similarly for companies constrained in (a). Your niche both protects and defines you.

Whereas if you want to start a startup, you're probably going to have to think of something fairly novel. A startup has to make something it can deliver to a large market, and ideas of that type are so valuable that all the obvious ones are already taken.

That space of ideas has been so thoroughly picked over that a startup generally has to work on something everyone else has overlooked. I was going to write that one has to make a conscious effort to find ideas everyone else has overlooked. But that's not how most startups get started. Usually successful startups happen because the founders are sufficiently different from other people that ideas few others can see seem obvious to them. Perhaps later they step back and notice they've found an idea in everyone else's blind spot, and from that point make a deliberate effort to stay there. [3] But at the moment when successful startups get started, much of the innovation is unconscious.

What's different about successful founders is that they can see different problems. It's a particularly good combination both to be good at technology and to face problems that can be solved by it, because technology changes so rapidly that formerly bad ideas often become good without anyone noticing. Steve Wozniak's problem was that he wanted his own computer. That was an unusual problem to have in 1975. But technological change was about to make it a much more common one. Because he not only wanted a computer but knew how to build them, Wozniak was able to make himself one. And the problem he solved for himself became one that Apple solved for millions of people in the coming years. But by the time it was obvious to ordinary people that this was a big market, Apple was already established.

Google has similar origins. Larry Page and Sergey Brin wanted to search the web. But unlike most people they had the technical expertise both to notice that existing search engines were not as good as they could be, and to know how to improve them. Over the next few years their problem became everyone's problem, as the web grew to a size where you didn't have to be a picky search expert to notice the old algorithms weren't good enough. But as happened with Apple, by the time everyone else realized how important search was, Google was entrenched.

That's one connection between startup ideas and technology. Rapid change in one area uncovers big, soluble problems in other areas. Sometimes the changes are advances, and what they change is solubility. That was the kind of change that yielded Apple; advances in chip technology finally let Steve Wozniak design a computer he could afford. But in Google's case the most important change was the growth of the web. What changed there was not solubility but bigness.

The other connection between startups and technology is that startups create new ways of doing things, and new ways of doing things are, in the broader sense of the word, new technology. When a startup both begins with an idea exposed by technological change and makes a product consisting of technology in the narrower sense (what used to be called "high technology"), it's easy to conflate the two. But the two connections are distinct and in principle one could start a startup that was neither driven by technological change, nor whose product consisted of technology except in the broader sense. [4]

Rate

How fast does a company have to grow to be considered a startup? There's no precise answer to that. "Startup" is a pole, not a threshold. Starting one is at first no more than a declaration of one's ambitions. You're committing not just to starting a company, but to starting a fast growing one, and you're thus committing to search for one of the rare ideas of that type. But at first you have no more than commitment. Starting a startup is like being an actor in that respect. "Actor" too is a pole rather than a threshold. At the beginning
of his career, an actor is a waiter who goes to auditions. Getting work makes him a successful actor, but he doesn't only become an actor when he's successful.

So the real question is not what growth rate makes a company a startup, but what growth rate successful startups tend to have. For founders that's more than a theoretical question, because it's equivalent to asking if they're on the right path.

The growth of a successful startup usually has three phases:

1. There's an initial period of slow or no growth while the startup tries to figure out what it's doing.
2. As the startup figures out how to make something lots of people want and how to reach those people, there's a period of rapid growth.
3. Eventually a successful startup will grow into a big company. Growth will slow, partly due to internal limits and partly because the company is starting to bump up against the limits of the markets it serves.

Together these three phases produce an S-curve. The phase whose growth defines the startup is the second one, the ascent. Its length and slope determine how big the company will be.

The slope is the company's growth rate. If there's one number every founder should always know, it's the company's growth rate. That's the measure of a startup. If you don't know that number, you don't even know if you're doing well or badly.

When I first meet founders and ask what their growth rate is, sometimes they tell me "we get about a hundred new customers a month." That's not a rate. What matters is not the absolute number of new customers, but the ratio of new customers to existing ones. If you're really getting a constant number of new customers every month, you're in trouble, because that means your growth rate is decreasing.

During Y Combinator we measure growth rate per week, partly because there is so little time before Demo Day, and partly because startups early on need frequent feedback from their users to tweak what they're doing. [6]

A good growth rate during YC is 5-7% a week. If you can hit 10% a week you're doing exceptionally well. If you can only manage 1%, it's a sign you haven't yet figured out what you're doing.

The best thing to measure the growth rate of is revenue. The next best, for startups that aren't charging initially, is active users. That's a reasonable proxy for revenue growth because whenever the startup does start trying to make money, their revenues will probably be a constant multiple of active users. [7]

Compass

We usually advise startups to pick a growth rate they think they can hit, and then just try to hit it every week. The key word here is "just." If they decide to grow at 7% a week and they hit that number, they're successful for that week. There's nothing more they need to do. But if they don't hit it, they've failed in the only thing that mattered, and should be correspondingly alarmed.

Programmers will recognize what we're doing here. We're turning starting a startup into an optimization problem. And anyone who has tried optimizing code knows how wonderfully effective that sort of narrow focus can be. Optimizing code means taking an existing program and changing it to use less of something, usually time or memory. You don't have to think about what the program should do, just make it faster. For most programmers this is very satisfying work. The narrow focus makes it a sort of puzzle, and you're generally surprised how fast you can solve it.

Focusing on hitting a growth rate reduces the otherwise bewilderingly multifarious problem of starting a startup to a single problem. You can use that target growth rate to make all your decisions for you; anything that gets you the growth you need is ipso facto right. Should you spend two days at a conference? Should you hire another programmer? Should you focus more on marketing? Should you spend time courting some big customer? Should you add x feature? Whatever gets you your target growth rate. [8]

Judging yourself by weekly growth doesn't mean you can look no more than a week ahead. Once you experience the pain of missing your target one week (it was the only thing that mattered, and you failed at it), you become interested in anything that could spare you such pain in the future. So you'll be willing for example to hire another programmer, who won't contribute to this week's growth but perhaps in a month will have implemented some new feature that will get you more users. But only if (a) the distraction of hiring someone won't make you miss your numbers in the short term, and (b) you're sufficiently worried about whether you can keep hitting your numbers without hiring someone new.

It's not that you don't think about the future, just that you think about it no more than necessary.

In theory this sort of hill-climbing could get a startup into trouble. They could end up on a local maximum. But in practice that never happens. Having to hit a growth number every week forces founders to act, and acting versus not acting is the high bit of succeeding. Nine times out of ten, sitting around strategizing is just a form of procrastination. Whereas founders' intuitions about which hill to climb are usually better than they realize. Plus the maxima in the space of startup ideas are not spiky and isolated. Most fairly good ideas are adjacent to even better ones.

The fascinating thing about optimizing for growth is that it can actually discover startup ideas. You can use the need
for growth as a form of evolutionary pressure. If you start out with some initial plan and modify it as necessary to keep hitting, say, 10% weekly growth, you may end up with a quite different company than you meant to start. But anything that grows consistently at 10% a week is almost certainly a better idea than you started with.

There’s a parallel here to small businesses. Just as the constraint of being located in a particular neighborhood helps define a bar, the constraint of growing at a certain rate can help define a startup.

You’ll generally do best to follow that constraint wherever it leads rather than being influenced by some initial vision, just as a scientist is better off following the truth wherever it leads rather than being influenced by what he wishes were the case. When Richard Feynman said that the imagination of nature was greater than the imagination of man, he meant that if you just keep following the truth you’ll discover cooler things than you could ever have made up. For startups, growth is a constraint much like truth. Every successful startup is at least partly a product of the imagination of growth. [9]

Value

It’s hard to find something that grows consistently at several percent a week, but if you do you may have found something surprisingly valuable. If we project forward we see why.

weekly yearly
1% 1.7x
2% 2.8x
5% 12.6x
7% 33.7x
10% 142.0x

A company that grows at 1% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x.

A company that grows at 1% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x. A company making $1000 a month (a typical number early in YC) and growing at 1% a week will 4 years later be making $7900 a month, which is less than a good programmer makes in salary in Silicon Valley. A startup that grows at 5% a week will grow 1.7x a year, whereas a company that grows at 5% a week will grow 12.6x.

Our ancestors must rarely have encountered cases of exponential growth, because our intuitions are no guide here. What happens to fast growing startups tends to surprise even the founders.

Small variations in growth rate produce qualitatively different outcomes. That’s why there’s a separate word for startups, and why startups do things that ordinary companies don’t, like raising money and getting acquired. And, strangely enough, it’s also why they fail so frequently.

Considering how valuable a successful startup can become, anyone familiar with the concept of expected value would be surprised if the failure rate weren’t high. If a successful startup could make a founder $100 million, then even if the chance of succeeding were only 1%, the expected value of starting one would be $1 million. And the probability of a group of sufficiently smart and determined founders succeeding on that scale might be significantly over 1%. For the right people—e.g. the young Bill Gates—the probability might be 20% or even 50%. So it’s not surprising that so many want to take a shot at it. In an efficient market, the number of failed startups should be proportionate to the size of the successes. And since the latter is huge the former should be too. [11]

What this means is that at any given time, the great majority of startups will be working on something that’s never going to go anywhere, and yet glorifying their doomed efforts with the grandiose title of “startup.”

This doesn’t bother me. It’s the same with other high-beta vocations, like being an actor or a novelist. I’ve long since gotten used to it. But it seems to bother a lot of people, particularly those who’ve started ordinary businesses. Many are annoyed that these so-called startups get all the attention, when hardly any of them will amount to anything.

If they stepped back and looked at the whole picture they might be less indignant. The mistake they’re making is that by basing their opinions on anecdotal evidence they’re implicitly judging by the median rather than the average. If you judge by the median startup, the whole concept of a startup seems like a fraud. You have to invent a bubble to explain why founders want to start them or investors want to fund them. But it’s a mistake to use the median in a domain with so much variation. If you look at the average outcome rather than the median, you can understand why investors like them, and why, if they aren’t median people, it’s a rational choice for founders to start them.

Deals

Why do investors like startups so much? Why are they so hot to invest in photo-sharing apps, rather than solid money-making businesses? Not only for the obvious reason.

The test of any investment is the ratio of return to risk. Startups pass that test because although they’re appallingly risky, the returns when they do succeed are so high. But that’s not the only reason investors like startups. An ordinary slower-growing business might have just as good a ratio of return to risk, if both were lower. So why are VCs interested only in high-growth companies? The reason is that they get paid by getting their capital back, ideally after the startup IPOs, or failing that when it’s acquired.

The other way to get returns from an investment is in the form of dividends. Why isn't there a parallel VC industry that invests in ordinary companies in return for a percentage of their profits? Because it's too easy for people who control a
private company to funnel its revenues to themselves (e.g. by buying overpriced components from a supplier they control) while making it look like the company is making little profit. Anyone who invested in private companies in return for dividends would have to pay close attention to their books.

The reason VCs like to invest in startups is not simply the returns, but also because such investments are so easy to oversee. The founders can't enrich themselves without also enriching the investors. [12]

Why do founders want to take the VCs' money? Growth, again. The constraint between good ideas and growth operates in both directions. It's not merely that you need a scalable idea to grow. If you have such an idea and don't grow fast enough, competitors will. Growing too slowly is particularly dangerous in a business with network effects, which the best startups usually have to some degree.

Almost every company needs some amount of funding to get started. But startups often raise money even when they are or could be profitable. It might seem foolish to sell stock in a profitable company for less than you think it will later be worth, but it's no more foolish than buying insurance. Fundamentally that's how the most successful startups view fundraising. They could grow the company on its own revenues, but the extra money and help supplied by VCs will let them grow even faster. Raising money lets you choose your growth rate.

Money to grow faster is always at the command of the most successful startups, because the VCs need them more than they need the VCs. A profitable startup could if it wanted just grow on its own revenues. Growing slower might be slightly dangerous, but chances are it wouldn't kill them. Whereas VCs need to invest in startups, and in particular the most successful startups, or they'll be out of business. Which means that any sufficiently promising startup will be offered money on terms they'd be crazy to refuse. And yet because of the scale of the successes in the startup business, VCs can still make money from such investments. You'd have to be crazy to believe your company was going to become as valuable as a high growth rate can make it, but some do.

Pretty much every successful startup will get acquisition offers too. Why? What is it about startups that makes other companies want to buy them? [13]

Fundamentally the same thing that makes everyone else want the stock of successful startups: a rapidly growing company is valuable. It's a good thing eBay bought Paypal, for example, because Paypal is now responsible for 43% of their sales and probably more of their growth.

But acquirers have an additional reason to want startups. A rapidly growing company is not merely valuable, but dangerous. If it keeps expanding, it might expand into the acquirer's own territory. Most product acquisitions have some component of fear. Even if an acquirer isn't threatened by the startup itself, they might be alarmed at the thought of what a competitor could do with it. And because startups are in this sense doubly valuable to acquirers, acquirers will often pay more than an ordinary investor would. [14]

Understand

The combination of founders, investors, and acquirers forms a natural ecosystem. It works so well that those who don't understand it are driven to invent conspiracy theories to explain how neatly things sometimes turn out. Just as our ancestors did to explain the apparently too neat workings of the natural world. But there is no secret cabal making it all work.

If you start from the mistaken assumption that Instagram was worthless, you have to invent a secret boss to force Mark Zuckerberg to buy it. To anyone who knows Mark Zuckerberg that is the reductio ad absurdum of the initial assumption. The reason he bought Instagram was that it was valuable and dangerous, and what made it so was growth.

If you want to understand startups, understand growth. Growth drives everything in this world. Growth is why startups usually work on technology—because ideas for fast growing companies are so rare that the best way to find new ones is to discover those recently made viable by chance, and technology is the best source of rapid change. Growth is why it's a rational choice economically for so many founders to try starting a startup: growth makes the successful companies so valuable that the expected value is high even though the risk is too. Growth is why VCs want to invest in startups: not just because the returns are high but also because generating returns from capital gains is easier to manage than generating returns from dividends. Growth explains why the most successful startups take VC money even if they don't need to: it lets them choose their growth rate. And growth explains why successful startups almost invariably get acquisition offers. To acquirers a fast-growing company is not merely valuable but dangerous too.

It's not just that if you want to succeed in some domain, you have to understand the forces driving it. Understanding growth is what starting a startup consists of. What you're really doing (and to the dismay of some observers, all you're really doing) when you start a startup is committing to solve a harder type of problem than ordinary businesses do. You're committing to search for one of the rare ideas that generates rapid growth. Because these ideas are so valuable, finding one is hard. The startup is the embodiment of your discoveries so far. Starting a startup is thus very much like deciding to be a research scientist: you're not committing to solve any specific problem; you don't know for sure which problems are soluble; but you're committing to try to discover something no one knew before. A startup founder is in effect an economic research scientist. Most don't discover anything that remarkable, but some discover

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208) 126 / 412
If the startup is taking the Facebook/Twitter route and the company that could start this way probably should. Evolutionary pressure is such a valuable technique that any launching something small and then using growth rate as a database will probably not do that. On the other hand, a startup building a new sharding it.

It may be that some of these limits could be overcome by changing the shape of the organization—specifically by sharding it.

This is, obviously, only for startups that have already launched or can launch during YC. A startup building a new database will probably not do that. On the other hand, launching something small and then using growth rate as evolutionary pressure is such a valuable technique that any company that could start this way probably should.

Notes

[1] Strictly speaking it's not lots of customers you need but a big market, meaning a high product of number of customers times how much they'll pay. But it's dangerous to have too few customers even if they pay a lot, or the power that individual customers have over you could turn you into a de facto consulting firm. So whatever market you're in, you'll usually do best to err on the side of making the broadest type of product for it.

[2] One year at Startup School David Heinemeier Hansson encouraged programmers who wanted to start businesses to use a restaurant as a model. What he meant, I believe, is that it's fine to start software companies constrained in (a) in the same way a restaurant is constrained in (b). I agree. Most people should not try to start startups.

[3] That sort of stepping back is one of the things we focus on at Y Combinator. It's common for founders to have discovered something intuitively without understanding all its implications. That's probably true of the biggest discoveries in any field.

[4] I got it wrong in "How to Make Wealth" when I said that a startup was a small company that takes on a hard technical problem. That is the most common recipe but not the only one.

[5] In principle companies aren't limited by the size of the markets they serve, because they could just expand into new markets. But there seem to be limits on the ability of big companies to do that. Which means the slowdown that comes from bumping up against the limits of one's markets is ultimately just another way in which internal limits are expressed.

[6] This is, obviously, only for startups that have already launched or can launch during YC. A startup building a new database will probably not do that. On the other hand, launching something small and then using growth rate as evolutionary pressure is such a valuable technique that any company that could start this way probably should.

[7] If the startup is taking the Facebook/Twitter route and building something they hope will be very popular but from which they don't yet have a definite plan to make money, the growth rate has to be higher, even though it's a proxy for revenue growth, because such companies need huge numbers of users to succeed at all.

Beware too of the edge case where something spreads rapidly but the churn is high as well, so that you have good net growth till you run through all the potential users, at which point it suddenly stops.

[8] Within YC when we say it's ipso facto right to do whatever gets you growth, it's implicit that this excludes trickery like buying users for more than their lifetime value, counting users as active when they're really not, bleeding out invites at a regularly increasing rate to manufacture a perfect growth curve, etc. Even if you were able to fool investors with such tricks, you'd ultimately be hurting yourself, because you're throwing off your own compass.

[9] Which is why it's such a dangerous mistake to believe that successful startups are simply the embodiment of some brilliant initial idea. What you're looking for initially is not so much a great idea as an idea that could evolve into a great one. The danger is that promising ideas are not merely blurry versions of great ones. They're often different in kind, because the early adopters you evolve the idea upon have different needs from the rest of the market. For example, the idea that evolves into Facebook isn't merely a subset of Facebook; the idea that evolves into Facebook is a site for Harvard undergrads.

[10] What if a company grew at 1.7x a year for a really long time? Could it not grow just as big as any successful startup? In principle yes, of course. If our hypothetical company making $1000 a month grew at 1% a week for 19 years, it would grow as big as a company growing at 5% a week for 4 years. But while such trajectories may be common in, say, real estate development, you don't see them much in the technology business. In technology, companies that grow slowly tend not to grow as big.

[11] Any expected value calculation varies from person to person depending on their utility function for money. I.e. the first million is worth more to most people than subsequent millions. How much more depends on the person. For founders who are younger or more ambitious the utility function is flatter. Which is probably part of the reason the founders of the most successful startups of all tend to be on the young side.

[12] More precisely, this is the case in the biggest winners, which is where all the returns come from. A startup founder could pull the same trick of enriching himself at the company's expense by selling them overpriced components. But it wouldn't be worth it for the founders of Google to do that. Only founders of failing startups would even be tempted, but those are writeoffs from the VCs' point of view anyway.
[13] Acquisitions fall into two categories: those where the acquirer wants the business, and those where the acquirer just wants the employees. The latter type is sometimes called an HR acquisition. Though nominally acquisitions and sometimes on a scale that has a significant effect on the expected value calculation for potential founders, HR acquisitions are viewed by acquirers as more akin to hiring bonuses.

[14] I once explained this to some founders who had recently arrived from Russia. They found it novel that if you threatened a company they'd pay a premium for you. "In Russia they just kill you," they said, and they were only partly joking. Economically, the fact that established companies can't simply eliminate new competitors may be one of the most valuable aspects of the rule of law. And so to the extent we see incumbents suppressing competitors via regulations or patent suits, we should worry, not because it's a departure from the rule of law per se but from what the rule of law is aiming at.

Thanks to Sam Altman, Marc Andreessen, Paul Buchheit, Patrick Collison, Jessica Livingston, Geoff Ralston, and Harj Taggar for reading drafts of this.

Lecture 6: Chasing Facebook’s Next Billion Users

http://www.businessweek.com/articles/2012-07-25/chasing-facebooks-next-billion-users

Later this year, Facebook (FB) will hit the 1 billion user mark. To casual observers, the spread of Mark Zuckerberg’s social network site across continents appears effortless. Yet for the past five years, the company has relied on a dedicated team charged with bringing new members into Facebook and getting them hooked. A kind of special-ops unit with influence over nearly every area of Facebook’s business, this growth squad has swelled to 150 people, from a founding group of five.

Core to this group is product manager Naomi Gleit, the second-longest-tenured Facebook employee. (Zuckerberg is first.) Gleit describes her team’s mission as making “Facebook available to everyone in the world.” Nearly half of the Internet’s worldwide 2 billion-plus population visits Facebook at least once a month. That’s a powerful draw for advertisers and makes life very difficult for social media up-and-comers (looking at you, Google+ and Twitter) that aspire to unseat the company as the world’s largest online network.

Keeping global expansion in the fast lane has taken on more urgency, given recent signs Facebook’s growth rates may be slowing down—and a big earnings disappointment on July 25 from Zynga (ZNGA), its largest game developer. The company’s U.S. users declined 1.1 percent over a six-month period ended in July, according to a research note published by CapStone Investments analyst Rory Maher on July 17. Since Facebook’s botched initial public offering in May, its shares have fallen more than 22 percent, to about $29 per share. (Facebook declined to comment on the CapStone research.)

Shooting to 1 Billion Users

![Graph showing Facebook's growth](https://www.businessweek.com/fileuploads/2012/07/25/chasing-facebooks-next-billion-users/1-billion-users-graph.png)

Graph by Bloomberg Businessweek Data: Facebook

http://startupclass.samaltman.com/ (gekregen van Inne ten Haveinne@darwine.nl 06-42804208) 128 / 412
Gleit joined Facebook in early 2005, when the site had just over 1 million users and fewer than 30 employees. She had become obsessed with the social network as an undergraduate at Stanford, writing a senior thesis about how Facebook beat out a rival website called Club Nexus on campus. After getting turned down for a position as assistant to then-President Sean Parker, she eventually landed a job on the marketing team. “She wanted to work at Facebook more than anything,” says Matt Cohler, an early executive at the company who hired Gleit and is now a general partner at the venture capital firm Benchmark Capital.

Gleit helped lead early efforts to expand Facebook’s reach beyond Ivy League schools to other colleges, followed by high schools, businesses, and, in 2006, to basically every living organism on the planet. “My job has been, over the years, to remove barriers that are preventing people from joining Facebook,” says Gleit, who turned 29 in July.

The growth team was formed in late 2007, when Zuckerberg decided expansion was so important that it warranted a unit with its own resources. The site was approaching 100 million members, but its growth rate had cooled. Led by former AOL (AOL) executive Chamath Palihapitiya, the original team included Microsoft (MSFT) engineering alum James Wang; Javier Olivan, who had founded a small social network in Spain; marketing whiz Alex Schultz; and Blake Ross, a programmer who had helped create the Firefox Web browser. To lead product management for the group, Zuckerberg tapped Gleit. “She is someone that the engineers both tremendously respected and also feared,” says Palihapitiya, who left Facebook last year. “We used to call her the Mom.”

The group was challenged by Zuckerberg to multiply the site’s user base. To accomplish this, the growth team struck a deal with Google (GOOG) to let the search engine show Facebook profiles in its results. They also launched a feature called “People You May Know,” which helped new users build their social network by showing them cousins, former classmates, and others they may have connections to on the site. Gleit and Olivan led a companywide push to create a translation tool that let users in Spain, France, and Germany navigate the site in their native languages. Within two years of its creation, the team had expanded Facebook’s roster of users sevenfold, to 360 million.

When Palihapitiya left, he handed leadership of the growth team to Olivan, who sits in Building 17 of Facebook’s new Menlo Park campus, across a walkway from the office occupied by Zuckerberg and COO Sheryl Sandberg. In an open workspace on the second floor, international flags line the rows of desks and computer dashboards display a running tally of monthly active users. “That’s the first thing we look at every morning,” Olivan says of the monitors. Recently, his team was renamed GEM—an acronym for growth, engagement, and mobile. Facebook’s new priority is keeping existing members active and logging in from mobile devices.

Gleit, who now runs her own team of product managers within GEM, says a key to building more active members is spotting what she calls “magic moments.” That’s when a new user moves from thinking, “What the hell is this Facebook thing all about” to “Aha! I understand, this is cool.” says Gleit. Facebook tries to get users to experience this moment as early as possible by helping them find friends effortlessly. Gleit helped develop functions like the contact importer, which suggests friends based on names in a user’s e-mail address book. Often, her job entails wrangling with other teams at Facebook to highlight features on the site that improve engagement. She describes her role as growth “evangelist.”

Facebook’s biggest opportunity to add new members is in emerging markets, where the social network isn’t always widely used. “The next billions of people, we believe, are going to come through mobile,” Olivan says. Gleit traveled to Israel in 2011 to help put together the buyout of Snaptu, which designs software that helps mobile-phone users access Facebook.

“There are still a lot of barriers in international markets, whether that be governmental barriers like in China, and occasionally Vietnam, or competitive barriers like Vkontakte in Russia,” she says, referring to the Eastern European social network that far eclipses Facebook, with more than 100 million users in that country.

As with many good ideas in Silicon Valley, Facebook’s concept of dedicating an entire business unit to growth has been replicated by other tech startups. Question-and-answer site Quora, online storage company Dropbox, and Twitter now have their own growth teams. The proof of concept will be the billion-member-mark celebrations sure to take place later this year at Facebook’s headquarters. Gleit’s more interested in looking ahead. After all, “1 billion is one-seventh of the world’s population,” she says.
Lecture 6: The only metric that matters

https://medium.com/@joshelman/the-only-metric-that-matters-ab24a585b5ea

Josh Elman

I've been lucky to be part of the early growth of several really interesting and now important networks including LinkedIn, Facebook, and Twitter. One of the things that I felt working on each of these is that we never looked at numbers or metrics in the abstract -- total page views, logged in accounts, etc, but we always talked about users. More specifically, what they were doing and why they were doing it. At LinkedIn we didn't talk about "total page views", but instead "profile views" - how many people were using LinkedIn to search for and find other people, and how many people were on LinkedIn being viewed. At Twitter while we had (and they still have) crazy page view numbers, we talked instead about how many people were looking at their timeline and reading tweets or tweeting.

When I meet new companies today, I often hear things like "We have 10M uniques with 30M page views per month." Or "We have a 25% DAU/MAU with a 2-cent ARPDAU". All of these sound pretty interesting in the abstract. But when you really dig in you start to find out nearly all of the 10M uniques all come in from search engines, click on 2-3 pages, and never come back. Or that the 25% DAU/MAU is because the app is spammy and drawing new users in very quickly but hardly any come back for a second session and no one ever pays.

While big numbers are a nice signal of, well, big numbers, I don't think they are an indicator at all for whether a product is really working. Whenever I hear some of these stats, I always ask the same question:

How many people are really using your product?

You need a metric that specifically answers this. It can be "x people did 3 searches in the past week". Or "y people visited my site 9 times in the past month". Or "z people made at least one purchase in the last 90 days." But whatever it is, it should be a signal that they are using their product in the way you expected and that they use it enough so that you believe they will come back to use it more and more. Once you can define a metric to answer this, then you can really track your growth on a day-to-day, week-over-week, month-over-month basis. And from there, you can identify the key supporting metrics that show you how likely it is more people will convert to using your product on a frequent basis, how likely they are to stay on your product vs churn out, etc.

At Twitter, we found that if you visited Twitter at least 7 times in a month, then it was likely you were going to be visiting Twitter in the next month, and the next month, and the next month. And we decided this was enough initially to be "really using it", though of course I think Twitter gets even better when people use Twitter every day or more.

More about the next most important metric -- what data shows you that more people will be using your product in the future, and how to use this focus to then refine your roadmap to get more people converted to using the product in another post.
7. How to build products users love
Lecture 7: How to Build Products Users Love

http://startupclass.samaltman.com/courses/lec07/

Kevin Hale

Alright, so when I talk about making products that users love, what it means specifically is "How do we make things that have a passionate user base, that our users are unconditionally wanting it to be successful, both on the products that we built and the companies behind them?" We're going to go over tons of information; try not to take too many notes - mostly just try to listen. I'll post a link to the slides on my Twitter account, and on that link, there will be a way for you to annotate the slides. So you can ask me questions, and if we don't get to them, I'll answer them after the talk.

So you guys have been listening to a lot about growth over the last several weeks, and to me, I feel like growth is fairly simple. It's the interaction between two concepts or variables: conversion rate and churn. The gap between those two things pretty much indicates how fast you're going to grow. Most people, especially business-type people, tend to look at this interaction in a very mathematical, calculated sort of way. Today I want to talk about these things at a more human scale because in a startup when you're interacting with your users, you have a fairly intimate interaction in the early stages, and so I think there's a different way of looking at this stuff in terms of how we build our products. We'll look at a lot of different examples of that and how it's executed well.

My philosophy behind a lot of things that I teach in startups is, the best way to get to $1 billion is to focus on the values that help you get that first dollar to acquire that first user. If you get that right, everything else will take care of itself. It's a sort of faith thing.

I came to be a partner at YC by a way of being an alumni. I went to the program of Winter 2006 (it was the second-ever program), and I built a product called Wufoo. Wufoo is an online form builder that helps you create contact forms, online surveys, and simple payment forms. It's basically a database app that looks like it's designed by Fisher-Price. What's interesting though is that because it was fairly easy to use, we had customers from every industry market and vertical you can think of including a majority of the Fortune 500 companies.

I ran the company for five years, and then we were acquired by Survey Monkey in 2013. At the time, we were a very interesting acquisition. We were only a team of 10 people at the time, and while we acquired funding here in Silicon Valley through Y Combinator, we actually ran the company from Florida. We had no office, everyone worked from home, and we were an interesting outlier. So each dot here represents a startup (PowerPoint slide) that exited through IPO or acquisition, we are the outlier to the left. The bottom represents the funding amount that they took, and vertical axis is the valuation of the company at the time. To sum it up the average start up raises about $25 million, and the return for their investors is about 676%. Wufoo, raised about $118,000 total, and our return to our investors was about 29,561%.

So a lot of people are interested in what makes Wufoo a little bit different, or how do we run the company differently. And a lot of it was focused on product. We weren't interested in building software that people just wanted to use, that reminded you that you worked in a cubicle because it was a database app at its core. We wanted a product that people wanted to love, that people wanted to have a relationship with, and we were actually very fanatical about how we approached this idea, to the point where it was almost sort of in a science-y way. So what we said was like, "What's interesting about startups in terms of us wanting to create things that people love, is that love and unconditional
feelings, are difficult things for us to do in real life. In startups, we have to do it at scale." So we
decided to start off by asking, “How do relationships work in the real world and how can we apply them
to the way we run our business and build our product that way?”

We'll go over these two metaphors: acquiring new users as if we are trying to date them, and existing
users as if they are a successful marriage.

When it comes to dating, a lot of the things that we uncovered, had to do with first impressions. All of
you often talk about your relationships in the origin story. You guys will tell me about your first kiss,
how you met, how you proposed. These are the things that we say over and over again, basically the
word-of-mouth stories for relationships. There are similar things that we do with companies. Human
beings are relationship-manufacturing creatures. We cannot help but create, and anthropomorphize, the
things we interact with over and over again. Whether it's the cars we drive, or the clothes we wear, or
the tools and softwares we use, we eventually prescribe characteristics to it, a personality, and we
expected it to behave a certain way - that's how we sort of interact with it.

First impressions are important for the start of any relationship because it's the one we tell over and
over again, right? There's something special about how we regard that origin story. Let me give you an
example. If you're on the first date with somebody, and you're having a nice dinner, but you catch them
picking their nose, you are probably not going to have another date with them. But if you're married to
someone for about 20 to 30 years, and you catch them on the BarcaLounger digging for gold, you don't
immediately call your lawyer, you know what I mean, and say, "We have a problem here, you have to
start drawing up papers for divorce." You shrug your shoulders, and say, "At least he has a heart of
gold."

So something about first-time interactions means that the threshold was so much lower in terms of pass
fail. So in software and for most products in Internet software that we use, first impressions are pretty
obvious and there are things you see a lot of companies pay attention to in terms of what they send
their marketing people to work on. My argument for people who are very good at product is that they
discover so many other first moments, and they make those something memorable: the first email you
ever get, what happens when you got your first login, the links, the advertisements, the very first time
you interacted with customer support. All of those are opportunities to seduce.

So how do we think about making first moments? We actually took this concept from the Japanese. They
actually have two words for how to describe things when you're finished with them, in terms of saying,
"is this a quality item?" The two words for quality are atarimae hinshitsu and miryokuteki hinshitsu. The
first one means taken for granted quality, which basically means functionality. The last one means
enchanting quality. Take for example a pen. Something has miryokuteki if the weight of the pen, the
way the ink flows out of it, the way it's viewed by the people reading the hand writing from the pen, is
pleasurable both to the user of the pen and the people who experience the byproducts, taking it to the
next level. Let's start with some examples.

This is Wufoo's login link, and it has a dinosaur on it, which I think is awesome! But if you hover it, the
spec has the added benefit of having a tool tip that doesn't tell you how to log in or what it does, but
basically "RARRR!" What we noticed about this in early usability stages, is that this put a smile on
people's faces, like hands down, universally. I think a lot of times when we are assessing products we
never think about, "Hey, what is the emotion on the person's face when they interact with this?"

This is Vimeo's launching page; this is actually a couple iterations ago, it's the one that I find to be the
most beautiful. It lets you know that when you're starting out on this journey with them, it is going to
be something different - they do this all over. If you search for the word "fart," as you scroll up and
down, it makes fart noises. There's something different, like this site interacts with you, it's a little bit magical, and it's a little bit different. It's something that you want to talk about.

You don't always have to do it with design. This is a sign-up form for Cork'd, which used to be a social network for people that love to drink wine. On it, it says, "Email address- it's also your sign in name, and has to be legit. First name - what mom calls you. Last name - what your army buddies call you. Password – something you'll remember but hard to guess. Password confirmation – type it again, think of it as a test." It's literally a poem as you fill it out the form. And this is the kind of thing when you're like, "Oh, I like the people behind this, I'm going to enjoy this experience." Now what does it say, when you fill out a form like this, about what their personality looks like it's going to be? And what's disappointing to me is Yahoo forces every product and service under them to use this exact same login form.

Flickr I thought had one of the best call-to-actions. It was, "Get in there!"

This is for Heroku's signup page. I think this is an older version. What's remarkable about it is that what you start getting a feel for, is like scaling up the backend services, it's as easy as dragging up-and-down different knobs and levers. It looks fairly easy to scale.

This is for a room full of computer science people; I think you'll appreciate that. This is Chocolat, a code editor. And they only have one call to action: when the time limit is up, everything in terms of all the features is all the same, except we changed the fonts to Comic Sans, and what they're basically saying is "Hey, we know who our users are, who our real customers are. They're going to be the people who care about this."

This is Hurl, a website for checking HTTP requests, and sometimes the places where you get errors are opportunities for first moments. If you hit a 404, this is what you get: (Unicorn throwing up a rainbow.)

Often times what we do is we create really beautiful marketing materials, but when you actually need documentation, we sort of skimp out on design features. This is something that happens over and over again. A company that gets this right is MailChimp. What they did was they redesigned all of their help guides so that they looked like magazines covers, and overnight basically readership goes up on all these features, and customer support for these things that help people optimize emails, goes down.

Speaking of documentation, Stripe - what's interesting about an API company, is that there is no UX. The UX is actually just documentation, and there are opportunities even in documentation to sort of enchant and amaze. One of the things that I love about them is that their examples are wonderful, but if you log into the app, one of the things that is a super pain for most people is when you're doing most people's APIs, is grabbing your API credentials and keys. And what shocked us is that it says, "If you are logged into the app, we automatically put your API credentials into the examples, so you only have to copy and paste once, when trying to learn their API."

When Wufoo wanted to launch the third version of our API, we realized, "Okay finally this is good enough that we want people to build on top of it." We were trying to figure out how we launch this out to the world that sort of has our personality behind it, because a lot of people usually do things like a programming API contest that give out iPads and iPhones; it makes you look like everyone else. So at our company, one weird value we have is that our cofounders are big medieval nuts, and we would take everyone out to Medieval Times every single year on the anniversary and founding of the company. So we said we have to do something in that flavor. We contacted the guys at armor.com, and said, "Can you forge us a custom battle axe?" We said, if you win our programming contest, you would win one. The result is, people wanted to talk about this. People wanted to say they were working on this because
they wanted to say, "I am programming for a weapon." What's cool is we had over 25 different applications created for us, of quality and quantity that we could not have paid for on the budget and time that we had. We got things like an iPhone app, an Android app, a Wordpress Plugin, and all we did was change the way people talked about our origin story of how they interacted with one of our services.

I'm going to shorthand this by saying you should just subscribe to Little Big Details. It's basically tons of screenshots of software that just shows that they are doing it right and being conscientious of the user and the customers.

When it comes to long-term relationships, or marriages, the only research that we ended up having to read is the stuff done by John Gottman. He's been featured in "This American Life," Malcolm Gladwell's books, etc. He's a marriage researcher up in Seattle, and he has an interesting parlor trick that he can do. He can watch a video tape of a couple fight over some issue for 15 minutes, and predict within 85% accuracy rate, whether that couple will be together or not, or divorced, in four years. If he increases that video up to an hour and asks them to also talk about their hopes and dreams, that prediction rating goes up to 94%. They showed these same video tapes to marriage counselors, successfully married couples, sociologist, psychiatrist, priests, etc. They can't predict with random chance, whether people are going to be together or not.

So John Gottman understands something fundamental about how relationships work in the long term, and that basically how we fight even in the short term period can indicate the whole system and what it's going to look like. One of the surprising things he discovered is not that successfully married people don't fight at all; turns out, everybody fights and we all fight about the exact same things: money, kids, sex, time, and others ("Others" are things like jealousy and the in-laws.) To bring this around, you can actually attribute every single one of these to problems to things you see in customer support when you're building out your products, so Money - this costs too much, or I'm having trouble with credit cards. Kids - users' client. Sex - performance, how long you're up and how fast. Others - I said was jealousy or in-laws, so that's competition and partnerships, anything weird happening there, people are going to write to you about. And the reason I like to think about this in terms of customer support is that, in everyone's processing of a conversion funnel, customer support is a thing that happens in between every one of the steps; it's the reason why people don't make it further down there; it's the thing that prevents conversion from happening.

Now as we were thinking through all of these ideas, and as we were building up the company, we realized that there's a big problem with how everyone starts up their company or builds up their engineering teams. There's a broken feedback loop there. People are divorced from the consequences of their actions. This is a result from the natural evolution of how most companies get founded, especially by technical cofounders. Before launch, it is a time of bliss, Nirvana, and opportunity. Nothing that you do is wrong. By your hand, which you feel is like God, every line you write and every code you write feels perfect; it's genius to you. The thing that happens is after launch, reality sets in, and all these other tasks come in to play; things that we have to deal with. Now what technical cofounders want to do is get back to that initial state, so what we often see is the company starts siloing off these other things that makes a startup company real, and have other people do them. In our minds these other tasks are inferior, and we have other people in the company do them.

So for us, what we're trying to figure out is how we change software development so that we inject some values that we don't talk about enough, like responsibility, accountability, humility, and modesty. We call this SDD (Support Driven Development). It's a way of creating high-quality software, but it's super simple; you don't need a bunch of Post-it notes. All you have to do is make everyone do customer
support. What you end up having is you fix the feedback. The people who built the software are the ones supporting it, and you get all these nice benefits as a result.

One of them is support responsible developers and designers. When people built the stuff, they give the very best support. Now we are not the first people to think of this. Paul English was a big supporter of this in Kayak. What he did was install a red customer support phone line in the middle of the engineering floor, and it would just ring with customer support calls. People would often ask him "Why would you pay engineers $120,000 or more to do something that you can pay other people a fraction of." He said, "Well, after the second or third time that the phone rings, and the engineer get the same problem, they stop what they're doing, they fix the bug, and they stop getting phone calls about it." It's a way of having QA in a sort of nice, elegant solution.

Now, John Gottman talks about the reason that we often break up with one another is due to four major causes. They are warning signs. He calls them the Four Horsemen: criticism, contempt, defensiveness, stonewalling. Criticism is basically people starting to focus, not just on the specific issue at hand, but on the over arching issues like "You never listen to users" or "You never think about us" all the time. Contempt is when somebody is purposely trying to insult another person. Defensiveness is not trying to take accountability, or trying to make excuses for their actions. Stonewalling is basically shutting down. Stonewalling according to John Gottman, is one of the worst things we can do in a relationship. Often times we don't worry about these things in customer support, criticism or contempt. Defensiveness, you see this all the time especially in companies as they get older. But stonewalling, this is something I see happening with startups all the time. You get a bunch of customer support calls coming in, and you just think, "I don't need to answer, I don't need to respond." That act of not even getting back to them is one of the worst things you can do, and it's probably some of the biggest causes of churn in the early stages of startups.

This is how support worked out with Wufoo. When we were acquired we had about 500,000 users on the system, 5 million people used Wufoo forms and reports whether they knew it or not, and all those people got support from the same 10 people, and usually there was one person dedicated to support a day, for any shift. Resulting in about 400 issues a week, that's about 800 emails. But our response time from 9 AM to 9 PM was between 7 to 12 minutes, from 9 AM to midnight was an hour, and then on the weekend it would be no longer than 24 hours. We carried this up all the way up to the scale.

What a lot of people talk about and often forget about Airbnb, is how they did this interesting thing where they went up to New York, and offered professional photography, and the founders would go up there and actually take pictures of the people's apartments to help them sell more, focusing on the stories about conversion. What most people don't realize is, a lot of times when I saw Joe in the early times of Airbnb, he had a phone headset stuck to his head all the time because he was doing phone support nonstop.

Churn is the story we don't like to talk about. Airbnb's growth really started picking up when they figured out how to match capacity to the demand or the phone calls they were getting into their support system.

At Wufoo we actually constantly did experiments around support because we were so obsessed with it. One experiment we did was, we heard someone here do a talk about how there's a disconnect between the emotions that we have when we need help, and the content and the reaction we get from people when we get help to people, especially online, because they just don't see those nonverbal cues. So she said, unless there's face recognition on the web, we are just always going to be disconnected from our users. Our feeling was like, "We're not face recognition experts, but we think there's another way of getting empathy." So, as form builders we added a drop-down, and what we said was, "what's your
emotional state." Our hypothesis was nobody's going to fill this out; we thought this was going to be a lame experiment, but we'll see how it goes. It turned out that this field was filled out 75.8% of the time. The Browser Type drop-down field in comparison was filled out 78.1% of the time. So people were basically telling us, "For my technical support issue, how I feel about this problem is just as important as all the technical details you need to figure out in order to debug it."

We didn't prioritize things or triage things by emotion, so for the most part people didn't game the system. One of the interesting byproducts of it was that we noticed that people started being nicer to us. We went back and looked at the data, did some text analysis and realized when it comes to communicating with people over written words like email, there's only three ways in which you show strong emotions: exclamation marks, curse words, AND ALL CAPS. Sure enough, on all three of those metrics, they've gone down in the way people were talking to us in customer support. Once people had a simple outlet for their emotions, it made them a lot more rational, and a made our jobs much more pleasant as a result.

The other byproduct that is awesome is that you actually build better software when you do this – far better software. This is actually backed up by a whole bunch of research. Jared Spool, at User Interface Engineer (one of the biggest players in the space) says that there's a direct correlation to how much time we spend directly exposed to users and how good our designs get. He said it has to come in this specific way. It has to be a direct exposure. It can't be something where someone generates a report or through a graph. You have to be interacting with them in somewhat real time. It has to be a minimum of every six weeks, and it has to be for at least two hours; otherwise your software will get worse over time. Our developers, the people who are with Wufoo, are getting exposed to our users 4 to 8 hours every single week. What it does is that it changes the way you sort of build software.

Jared Spool has another way of talking about how we build products. Imagine that this represents all the knowledge needed to use your app on a spectrum (PowerPoint slide). This is like no knowledge (far-left) and this is all the knowledge needed (far-right). These two lines are pretty much your interaction with users. This is currently where their knowledge point is (PowerPoint slide), and this is the target point where you're trying to get them to. The gap between the two is called the knowledge gap as Spool calls it. And what's interesting about this is there's only two ways to fix this. That gap represents how intuitive your app is. You either get the user to increase their knowledge or decrease the amount of knowledge that's needed to use the application. And often times as engineers or people who build and work on these products we think let's add new features. New features only means let's increase the knowledge gap.

So for us we actually focused on the other direction. What that meant is that we spent 30% of our engineering time on internal tools to help with our customer support. But often times it was spent on helping people help themselves, like frequently asked questions, or tooltips; things like if you just click the help link, instead of taking you to the generic help documentation page, you go to the specific page that's going to be the most appropriate for what you're working on. We redesigned our documentation over and over again, A/B tested it constantly. One iteration of our documentation page reduced customer support by 30% overnight. It's one of those things where overnight, all the people that work on the product immediately had 30% less work to do.

What happens if you have everyone working on customer support constantly? I talked about in the very beginning that growth is a function of conversion and churn. This is Wufoo's growth curve for the first five years (PowerPoint slide). What's interesting is that we paid no money on advertising or marketing; all of it was done by word-of-mouth growth. The interaction between new users and downgrades are this (PowerPoint slide). It's so slight what it takes, that gap, making that work. What a lot of people keep forgetting is that there's almost no difference between an increase in conversion rate, 1% increase, and
1% decrease in churn; they do the exact same thing to your growth.; however, the latter is actually much easier to do, and much cheaper to do. And a lot of times we neglect this until way far along, and we usually have our B team work on these projects and services.

This is actually not one of the graphs we tracked most of the time at Wufoo, it's not even the one I'm proud of. This is the one I'm proud of because even though we had this nice, awesome curve of growth, this is what allowed us to scale, keep the company small, and have an awesome culture. And that required doing a lot of these things to help people do what they need.

So John Gottman noticed that there was a different type of behavior for relationships, and why people divorced. Basically there were these subsets of people who stayed together 10 to 15 years and all of a sudden divorced. None of the other indicators would show that this was going to happen. He was looking through the data and realized, "Oh there's no passion, there's no fire between these people." When it comes to relationships, they kind of follow the second law of thermodynamics: In an enclosed energy system, things tend to run down, so you constantly have to be putting energy and effort back into it. The way a lot of people think showing people that "I care about you" in products and companies is by doing things like creating a blog or making a newsletter. The thing is we look at these rates and basically it was such a small percentage of our active users, most of them had no idea about the awesome things that we were doing for them. So we built a new tool and called it the Wufoo Alert System. It allowed us to timestamp every new feature that we are building for users, and that every time they would login we would look at the difference between their log in time, or last login in time, and the new features that were implemented, and they would have this message show up, "Hey since you've been gone, here's all the awesome stuff that Wufoo did for you." Hands-down, this was the most talked about feature that I heard every time I went out to talk to users. They would say things like, "Dude I love that 'Since you've been gone' thing. Even though I pay the same amount every single month, you guys are doing something for me almost every week. It's totally awesome; it makes me feel like I'm getting maximum value."

The other thing that we did in addition to having everyone support the people that paid their paycheck, is have them say "thank you." And this was in large part due to us injecting humility and modesty into the equation. Every single Friday we would get together, we would write simple handwritten thank you cards to our users. And I know there are tons of people who would not be sort of excited about doing this; it was a ritual that made all the difference in terms of like having a team that was very tightly knit, and working on stuff that they really cared about. They constantly knew what the mission was for, and why we sort of did what we did. These aren't fancy thank you cards; they're just simple handwritten stuff on index cards, we threw in a sticker, and slapped on a dinosaur on the front of it.

What's interesting is we started this practice as a result of the early days of starting Wufoo. Chris, Ryan, and I were talking to try to figure out what we were going to do to show users that we appreciated them around Christmas, and Chris came up with this idea where he said, "Hey guys, a couple years ago my mom made me write thank you letters to all my relatives for my Christmas presents, and I really didn't like to do it, but the following year all my presents were super good... so I think we should try this for our business and see how it goes."

So that first year we wrote handwritten Christmas cards to all of our users that first year. Second year rolls around, and we have too many customers with just the three founders. We were thinking, "We're kind of screwed; we don't know what we're going to do." Well, we read a book called The Ultimate Question and in it, he talks about focusing on your most profitable users; if you just take care of them, things will work out. So we thought, "That will work out, that's scalable." Basically we only wrote to our highest-paying customers. So January rolls around that second year and one of our longtime loyal users writes to us. He basically says, "Hey guys I really loved the Christmas card you guys sent me the first
year, and I just wanted you to know, I haven't received my second card yet and I'm just looking forward to it; I know you didn't forget about me. Thanks a lot." So we were like, fuck. The best way to exceed expectations is not to set any in the beginning; we were sort of in this conundrum. What we decided after thinking about it for a while was that we had to stop doing it just one time a year; it needs to be something that happens every sort of week. And even though we'll never catch up to all of our customers, just a practice of doing it will make all the difference.

I talked a lot about lovey-dovey, touchy-feely stuff that I think a lot of engineers don't like to think about too often, so I'll end on hard business data or research. There's an article that was put out by the Harvard Business Review several years ago by Michael Treacy and Fred Wiersema and in it they talk about the discipline of market leaders. They say there's only three ways that you achieve market dominance, and depending on how you want to achieve that market dominance, you have to organize your company in a very specific way: best price, best product, and best overall solution. For best price, you focus on logistics, so Wal-Mart and Amazon. If you want to be the best product out there, you focus on R&D, quintessential example of that. Best overall solution is about being customer intimate. This is the path that you see all luxury brands follow, as well as the hospitality industry. What I love about this path towards market dominance is that the third one is the only one that everyone can do at any stage of their company. It requires almost no money to get started with it. It usually just requires a little bit of humility and some manners. And as a result, you can achieve the success of any other people in your market. That's all I got, thank you very much.

Q: So what do you do when you have a product with many different types of users? How do you build one product that all these users love?

A: There is an interesting fine line for that. What I usually tell people is focus on those who are the most passionate, especially in the early stages. Whatever niche it's going to be in, that's where I'm going to focus on completely. I think Ben Silverman from Pinterest started off with design bloggers. Tailor your thing for them, and eventually you'll figure out universal values that will appeal to a lot of other people. So just start one at a time. And a lot of examples that you see up there, a lot of companies make the mistake of just thinking "Oh I'll just make my app funny." Humor is like really difficult to do. When you want to shoot for something witty, you have to get functionality right. So like the Japanese quality. If you don't have atarimae, don't try to do anything witty, because it'll backfire. So hands down our number one focus in Wufoo was to make everything as easy to use as possible; everything else was just polish.

Q: How do we balance being obsessed with working on product with all the other skills that are needed by a company, such as marketing, branding, etc?

A: If you're working on product, you should also always have this flip-side where you're talking to users. For us inside of Wufoo, the way we got people to talk to users was through customer support. They got to see firsthand whether the features worked or not, and it also impacted everyone else in the company because everyone had a customer support shift, so they had a social incentive to make everything work. There should be no point where you are only focused on product. You should always have time where you work on product, and then you see what users say to you - like ongoing virtual feedback. So be careful when you don't have that.

My feeling on marketing and sales, my feeling is marketing and sales is a tax you pay because you haven't made your product remarkable. Word-of-mouth is the easiest kind of growth, and it's how a lot of the great companies grow. Figure out how to have a story that people want to tell about your product where they are the most interesting one at the dinner table. And then that person is your sales person. That person is your sales force for you.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Q: How do you make a decision on product and communicate that with your engineering team when there are lots of different directions to go?

A: We just looked at customer support, which is really easy because you see what people are having the most amount of trouble with. You cannot help but get feature requests from people. No matter whatever openings you have in your product or app, people will jam feature requests in there, so you're going to know what they want. Your job as a product person and an engineer is to not just do what they say, because that way you'll just be a slave. You have to figure out and solve what they really want, that deep underlying reason. The thing is if everyone wants to have a different way to go, then ultimately someone's going to figure something out. But also, make the smallest version of each little idea, no longer than 1 to 2 weeks to build it, so you can try it out to see what works and what doesn't. It's dangerous to have multiple product directions that require a lot of time to figure out.

Q: Can you relay the story about how the King for a Day thing was good at Wufoo?

A: Yeah, okay so I don't like hackathons. I think they sort of suck in terms of those done inside of companies because you spend like 48 hours working really hard on something that you're really passionate about, and 99% of them never make it out to production. It's super sad. So we came up with an idea called "King for a Day." It worked over the weekend. How it worked is someone randomly in the company got drawn and they got to be the king. The king got to tell everyone else what to do on product. So everything that was bothering them about Wufoo or any other feature that they wanted to have built, they got engineering, marketing, and advertising resources of everyone in the company to make it happen. And of course we worked with them to figure out what we could do in 48 hours. We would do this one to two times a year. It was a huge hit and a boost to morale because what people most loved, was working on things that they felt made a difference, like, I made a difference to the app. So for us, that's one way that we would divide time for product direction. Sometimes, the people that work for you are the people who have the strongest opinions about where the product should go. And that's a good way to democratize it a little bit, by rotating it around.

Q: You said you guys all work from home, which usually seems like a nightmare. How did you make that work?

A: We all work from home, and we all work around the Tampa Bay area. We would allow anyone to work from anywhere but usually as we tried to recruit them and meet our team, they usually decided to come and move here anyway. Remote working is especially tricky. A lot of people like to romanticize it, especially people who are employees, but the thing is an office gives you a lot of benefits and efficiencies that you now have to compensate for when you have remote working. But remote working also has these sort of efficiencies. For example, I don't have to worry about my employees losing two hours of their day to commuting. So the biggest thing we had to do for remote working is to respect people's time. The way we had it set up is we actually had a 4 1/2 day workweek at Wufoo; half-day on Friday was for all the meetings and stuff. We said, no biz dev meetings, no talking with other outside parties. They'd have to be done on Friday, on that half-day; they couldn't be done in the middle of the week. And then also one day of everyone was already dedicated to customer support. So everyone in our company effectively only had three days each week to actually build and work on whatever they were doing. But I actually firmly believe that if you have three solid days, 8 to 10 hours, when you're only working on what you need to build, you can get a ton of shit done. So, what we said was, you have to respect everyone's time during that three day period.

What we came up with was a 15 minute rule. You could have a chat or a phone call with someone, but it could last no longer than 15 minutes. So if you had some complicated issue that you couldn't figure
out, at 15 minutes you are to immediately table that item, and have us discuss it on Friday. You'd move on to the next item on your list. I would say 90% of the time, the item never got brought up on Friday, because usually what would happen is people would sleep on it, and then you would magically say "Hey I found a solution!" Or "Hey that's not a big problem whatsoever." Most problems inside a company don't need to be solved in real time or right away. The only things are like when the site is down or when payments aren't working. Everything outside of that is kind of luxury. So focus on your priorities as much as possible, and as a result our 10 person team did far more than many many other companies.

But it takes extra work to make remote working happen. We are an extremely disciplined team, and I would have to say, there are not many YC companies that have been able to replicate what we do. I think there are only two companies in YC that have been able to replicate our discipline style. It takes more work in a very different fashion. And often allows you to be a little bit lazier, in terms of all these things around productivity.

Q: How do we set up accountability for our employees as a manager?

A: We were profitable nine months after launch, so we had profit-sharing, which makes incentives pretty simple and clear. It would be a multiple of whatever sort of bonus pool that we had, and performance measures would be based on how they did in customer support, on their duties there, and what they said they wanted to accomplish. I don't like process and I don't like a lot of tools to help people to be productive, so the only thing that we had to help people manage their projects is a To-do list. It was a simple text file that we shared in a Dropbox account. Each person had their name on it, and you got to see every time someone updated things on the To-do list. What we said was every single night, you wrote everything you did that day, and on Friday we would just go over "This is what you said last week you were going to do; this is what you actually got done. What are the problems at hand?" And it's super simple. It creates this nice written trail for how to handle stuff, and I don't have to worry about managing them. They set the tone for how they want to be assessed. And for people who are excellent at what they do, it works very very well. And when you actually have problems, it's very easy to fire people. I was fortunate enough not to have to fire anyone at Wufoo, but we were able to correct everyone's behavior very very quickly because we just looked at this and evaluated the problem: "Look this is a pattern of behavior. You've been doing your work at last minute, etc. This is evidence that you've provided to us; all we have to do is describe it back to you." And because everyone in the company sees it, there's social pressure that's put into place to help make it all happen.

Q: How do you hire people that can work remotely and work in this fashion?

A: Pretty easily, you have them work on a side project for you. So you contract them out, and have them work remotely as such. Usually the projects I like to have them work on are about one month long so you can get a good sense of how people manage themselves and work on things. That was always the first assessment; we never did things just by interviews.

The other thing we had to screen them for was their ability to do customer support because not every engineer has those empathy skills to handle that stress. So sometimes I would have people write "break-up" letters to me in an interview, giving them 15 minutes to write it. That way you get a good sense of their writing skills because 90% of what you're doing in customer support is telling customers bad news like "we don't support that feature, sorry," or "no that's not going to work," or "that's not going to be available."

Q: Are there any tricks or experiments that didn't work out in your company?
A: Okay I'll talk about one. So one of the things that we did early on to try to motivate ourselves was - like, we understood the idea of crunch mode, and that it's really bad for people. Like if you're doing the subscription business, you need people to last for the long-term, and in video games, a lot of the time they crunch people for a specific time and they have multiple sprints. Most of the time the deadlines can get super exhausting. You might get an increase in productivity, but the recovery that you need for people is always greater than the productivity you gain. And in a company where you need everyone doing customer support, being on their game, and constantly pushing out features, you don't have time for recovery.

So we were thinking that we wanted to build a company vacation into how Wufoo works to reward our users every single year. So we thought, if the vacation is built in for the recovery, we can have one crunch period before the vacation set up and only do customer support that will sort of scale with people. So the way we did the very first crunch mode was that, it was just between the three founders, and we had each of us draw a 10 item To-do list that would be fairly aggressive. The first person to get through seven of their items would win, and the last person to get through seven of their items, would become what we called "Trip Bitch." Trip Bitch meant that you carried the other person's luggage and got people drinks when you're on the company vacation. So we did that, and during that period, everyone was pretty excited about it. The winner also got to choose the next company vacation. But all of a sudden, Ryan had basically poorly estimated the items on his list and realized very quickly, "I'm going to fucking lose," and he just sort of gave up. So crunch mode, turned out to be blah mode for him because he knew he was going to lose and became pretty demoralized. So as a result of that we decided not to do it in that similar fashion anymore. Good idea that we like to talk about, but it was one that we never did again.

Alright guys, thanks a lot! You can email me at kevin@ycombinator.com.
Lecture 7: How to Build Products Users Love

http://seriouspony.com/blog/2013/7/24/your-app-makes-me-fat

The Experiment

1. “Memorize these numbers.”

<table>
<thead>
<tr>
<th>Group A</th>
<th>Group B</th>
</tr>
</thead>
<tbody>
<tr>
<td>two-digit memorization task</td>
<td>seven-digit memorization task</td>
</tr>
<tr>
<td>memorizing 2 numbers… 2, 7</td>
<td>memorizing 7 numbers… 2, 7, 6, 9, 1, 5, 8</td>
</tr>
</tbody>
</table>

In 1999, Professor Baba Shiv (currently at Stanford) and his co-author Alex Fedorikhin did a simple experiment on 165 grad students. They asked half to memorize a seven-digit number and the other half to memorize a two-digit number. After completing the memorization task, participants were told the experiment was over, and then offered a snack choice of either chocolate cake or a fruit bowl.

The participants who memorized the seven-digit number were nearly 50% more likely than the other group to choose cake over fruit.

Researchers were astonished by a pile of experiments that led to one bizarre conclusion:

Willpower and cognitive processing draw from the same pool of resources.

Spend hours at work on a tricky design problem? You’re more likely to stop at Burger King on the drive home. Hold back from saying what you really think during one of those long-ass, painful meetings? You’ll struggle with the code you write later that day.

Since both willpower/self-control and cognitive tasks drain the same tank, deplete it over here, pay the price over there.

The tank is empty.

And even if you loved solving tough puzzles at work, the drain on your self-control still happens. One pool. Whether the drain was from something you love or hate doesn’t matter.

Cognitive resource tank don’t care.

You snap at the kids or dog over the tiniest thing. Or the dog snaps at you.

The Dog Experiment

1. “Wait for 10 minutes.”

Group A
“Sit” (wait outside their crate)

Group B
Wait in their crate

An experiment asked one group of dogs to sit, just sit, nothing else, for a few minutes before being released to play with their favorite treat “puzzle” toy (the ones where the dog has to work at getting the treats out of it). The other group of dogs were allowed to just hang out in their crates before getting the treat puzzle.

You know where this goes: the dogs that had to sit — exercising self-control — gave up on the puzzle much earlier than the dogs that were just hanging out in their crate. The dogs that were NOT burning cognitive resources being
obedient had more determination and mental/emotional energy for solving the puzzle. Think about that next time you ask Sparky to be patient. *His cognitive resources are easily-depleted too.*

Now think about what we're doing to our users.

If your UX asks the user to make choices, for example, even if those choices are both clear and useful, the act of deciding is a cognitive drain. And not just while they're deciding... even after we choose, an unconscious cognitive background thread is slowly consuming/leaking resources, "Was that the right choice?"

If your app is confusing and your tech support / FAQ isn't helpful, you're drawing down my scarce, precious, cognitive resources. If your app behaves counter-intuitively – even just once – I'll leak cog resources *every time I use it, forever*, wondering, "wait, did that do what I expected?". Or let's say your app is super easy to use, but designed and tuned for persuasive brain hacks ("nudges", gamification, behavioral tricks, etc.) to keep me "engaged" for your benefit, not mine (lookin' at you, Zynga)... you've still drained my cognitive resources.

And when I back away from the screen and walk to the kitchen...

*Your app makes me fat.*

If our work drains a user’s cognitive resources, what does he lose? What else could he have done with those scarce, precious, easily-depleted resources? Maybe he's trying to stick with that diet. Or practice guitar. *Or play with his kids.*

That one new feature you added? That sparkly, Techcrunchable, awesome feature? What did it cost your user? If the result of your work consumes someone's cognitive resources, *they can't use those resources for other things that truly deeply matter.* This is *NOT* about consuming their time and attention while they're using your app. This is about draining their ability for logical thinking, problem-solving, and willpower after the clicking/swiping/gesturing is done.

Of course it's not implicitly bad if our work burns a user's cog resources. Your app might be the one place your user wants to spend those resources. But knowing that interacting with our product comes at a precious cost, maybe we'll make different choices.

Maybe we'll think more about what our users really care about. Maybe we'll ask ourselves at each design meeting, "is this a Fruit-choosing feature or a Cake-choosing feature?" and we'll try to limit Cake-choosing features—the ones that really drain them — to that which supports the thing they're using our app for in the first place.

(Yes, cognitive resources can be partly replenished throughout the day by getting glucose to the brain, but be careful with that. A high-protein snack combined with small infrequent sips on a sports drink can help, a lot.)

But even if we can justify consuming our user's cognitive resources while they're using our product, what about our *marketing*? Can we honestly believe that our "content marketing" is a good use of their resources? "Yes, because it adds value." we tell ourselves. But what does that even mean? Can we honestly say that "engaging with our brand" is a healthy, ethical use of their scarce, precious, limited cognitive resources? *Yes, because our content is useful.*

And that's all awesome and fabulous and social and 3.0ish except for one, small, inconvenient fact: **zero sum.** What you consume here, you take from there. Not just their attention, not just their time, but their *ability to be the person they are when they are at their best.* When they have ample cognitive resources. When they can think, solve-problems, and exercise self-control. When they can create, make connections, and stay focused.

Is that "content" worth it? Maybe. But instead of "Is this useful?" perhaps we should raise the bar and ask "Will they use it?" (and so, yeah, I'm more than a little self-conscious about typing that as I consume your cognitive resources. But I didn't start Serious Pony to save your cognitive resources; I want to help save the cognitive resources of your users).

I'm not against "content marketing". On the contrary, it's nearly the only form of cog-resource-draining marketing that can be "worth it". It's the one form of marketing that can help people become better at something they care about. It's one form of marketing with the potential to deliver the user-learning so few companies care about. Content marketing can (and should) be "the missing manual." It can (and should be) the inspiration for our users to learn, get better (at the thing they care about), and connect with other users.

But if it's "content" designed solely to suck people in ("7 ways to be OMG awesome!!") for the chance to "convert", we're *hurting* people. If we're pumping out "content" because frequency, we're *hurting* people. I'm hurting some of you now. That's on me. It's why I try to use graphics to make the key point, so you don't have to read the post (also because I'm really ramby-aroundy, I know, workin' on it.)

My father died unexpectedly last week, and as happens when one close to us dies, I had the "on their deathbed, nobody thinks..." moment. Over the past 20 years of my work, I've created interactive marketing games, gamified sites (before it was called that), and dozens of other projects carefully, artfully, scientifically designed to slurp (gulp) cognitive resources for... very little that was "worth it". Did people willingly choose to engage with them? Of course. And by "of course" I mean, *not really, no.* Not according to psychology, neuroscience, and behavioral economics.
research of the past 50 years. They were nudged/seduced/tricked. And I was pretty good at it. I am so very, very sorry.

My goal for Serious Pony is to help all of us take better care of our users. Not just while they are interacting with our app, site, product, but after. Not just because they are our users, but because they are people.

Because on their deathbed, our users won't be thinking, “If only I'd spent more time engaging with brands.”

Help them conserve and manage their scarce, precious, easily-depleted cognitive resources for what really matters. To them. And don’t forget to take care of your own. Think of the kids. Think of Sparky.

Lecture 7: What Makes a Design Seem 'Intuitive'?

http://www.uei.com/articles/design_intuitive/  

In a recent usability test, I once again witnessed something I've seen a hundred times before: a frustrated user claiming he knows exactly what is wrong with the interface he was fighting with. What was his suggestion? “These guys need to make this thing a lot more intuitive. The problem is that this program isn't intuitive enough. It needs to be more intuitive!”

I think he used the I-Word no less than 25 times during the session. His frustration was real and his desire was great. So, why wasn't the interface 'intuitive'? Well, it's probably because it's really, really hard to do.

People Intuit, not Interfaces

To those who police the English language, interfaces can't be intuitive, since they are the behavior side of programs and programs can't intuit anything. When someone is asking for an intuitive interface, what they are really asking for is an interface that they, themselves, can intuit easily. They are really saying, “I want something I find intuitive.”

But, I believe that English is an adaptable medium, so it's ok with me if we call a design intuitive. Yet, what does it mean, from a design standpoint, when someone desires a design to be intuitive?

To answer that question, we first have to look at how people understand the design in the first place. To do that, we need to look at the design's knowledge space.

Current and Target Knowledge Points

Imagine a long wall where you'll line up all the users who will use your design. We're going to want to organize the wall, so against the left side, we'll put everyone who knows absolutely nothing about how to use the interface. (Maybe they don't even know how to use a mouse.)

On the right side, we'll put everyone who knows everything there is to know about the design. (That may only be the designers.) We'll organize all the people along the wall by how much they know. If they know only a little, they'll stand closer to the left. The more they know, the closer we put them to the right. (Here is a picture of what our wall might look like:)

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
You can think of an interface’s knowledge space as a continuum which goes from knowing nothing about the interface and to knowing everything someone could possibly know.

If you’re looking at the wall, the distance from the left represents how much any given user knows about the design. For each user, we call this the current knowledge point. That’s the amount of knowledge they have when they approach the interface.

There’s another point that’s of interest to us: the target knowledge point. This point represents how much knowledge the user needs to know to accomplish their objective. Every time a specific user tries to complete a specific task, the current knowledge and target knowledge points become very important to us. (Here you’ll see our wall with sample current and target knowledge points marked off: )

For a given user trying to complete a given task with an interface, there are two points in the knowledge space that interest us most. Current Knowledge represents the knowledge the user has when they first approach the interface to complete the task. Target Knowledge is the knowledge the user needs to accomplish the task.

Now, every user will have a different current knowledge point and that point changes as they get more experience. Yet, we’ve found that, by plotting out different users, we often see very clear clusters–bunches of users that share extremely similar current knowledge. Working with users in the middle of several of the most important clusters gives design teams a nice place to start. (Using these clusters can help design teams determine which personas to focus on.)

The Knowledge Gap

The distance between current knowledge and target knowledge has a technical name: "The Gap". (Subsequently, an entire chain of clothing stores was named after it!)

The Knowledge Gap is where design happens. We don’t need to design to the left of current knowledge point, because it’s all stuff the user already knows. And we don’t need to design stuff to the right of the target knowledge point, since the user won’t be needing that information (for this task, at least). We only need to design the interface for the space in between current knowledge and target knowledge. (See a picture of the Knowledge Gap here: )

The space between the Current Knowledge and Target Knowledge points is called The Knowledge Gap. This is the portion of the knowledge space we’re most concerned with when we’re designing interfaces.

Users can complete their objective when current knowledge equals target knowledge. There are two ways this can happen. You can train the user, thereby increasing their current knowledge, until they know everything they need to know. Or, you can reduce the knowledge necessary, by making the interface easier, until target knowledge only requires the information the user already has. In fact, most good design involves both: users are trained (through explanatory text and other devices) while the designer reduces complexity, reducing the gap distance from both directions.

The Two Conditions of Intuitive

In our research, we’ve discovered that there are two conditions where users will tell you an interface seems ‘intuitive’ to them. It only takes meeting one of the two conditions to get the user to tell you the design is intuitive. When neither condition is met, the same user will likely complain that the interface feels ‘unintuitive’.

Condition #1:
Both the current knowledge point and the target knowledge point are identical. When the user walks up to the design, they know everything they need to operate it and complete their objective.

Condition #2:
The current knowledge point and the target knowledge point are separate, but the user is completely unaware the design is helping them bridge the gap. The user is being trained, but in a way that seems natural.

The Hotel Phone

Recently, I stayed in a hotel while visiting an old friend. Wanting to call my friend to warn him of my imminent arrival, I approached the phone in my hotel room and lifted the receiver, ready to make my call. Can you guess what button I pressed first?

Chances are you guessed the ‘9’ button. As adults, we learn at an early age that the ‘9’ button will get us an outside line when using a business or hotel phone system. This becomes part of our current knowledge as we travel from
phone system to phone system. '9' becomes intuitive, though it isn't innate&we had to learn it somewhere along the way.

Of course, for this hotel, you would've been wrong. The designers of this phone felt that the '8' button was a much better choice. How unintuitive could they be? Everybody knows '9' is far more intuitive!

Because other people had problems with this, there were little signs all over--on the phone, on the wall, on the receiver--that stated you needed to press '8' to get an outside line. I immediately saw these signs and, without really contemplating the design, pressed '8' and the rest of my friend's number.

The signs made '8' seem intuitive by training me without my even realizing it. They narrowed the gap quickly and without the distraction often associated with learning new things. Had the phone used the '9' button, it would've met condition #1. However, since it had the signs for the '8' button and they worked unobtrusively, it met condition #2.

Making Designs Seem Intuitive

The biggest challenge in making a design seem intuitive to users is learning where the current and target knowledge points are. What do users already know and what do they need to know? To build intuitive interfaces, answering these two questions is critical.

For identifying the user's current knowledge, we favor field studies. Watching potential users, in their own environments, working with their normal set of tools, and facing their daily challenges, gives us tremendous insight in what knowledge they will have and where the upper bounds are. Teams receive a wealth of valuable information with every site visit.

For identifying necessary target knowledge for important tasks, usability testing is a favorite technique of ours. When we sit users in front of a design, the knowledge gap becomes instantly visible. (We've had great success, right after a test, listing out all the knowledge the user needed to acquire during the test. It can be quite revealing!)

Intuitive Doesn't Always Make Sense

Unfortunately, making an interface intuitive often increases development costs dramatically. Reducing target knowledge, particularly for large knowledge gaps, can be a very expensive process, particularly if you have to build complex tools, such as wizards and data auditors.

Anyone who has tried to build a tool that reduces target knowledge knows that they take tremendous work to get right. Is making an interface intuitive worth the investment? Not always.

For example, Amazon makes the process of returning a purchased product fairly intuitive. Once a user finds the (sometimes hidden) magic button on the order form, they have no trouble going through the return process--a multi-step wizard which asks intelligent questions and guides the user through the process of printing a shipping label, determining the shipping costs, and returning the product.

However, in our studies, users have much more difficulty finding a phone number to call Amazon's customer service center. Amazon doesn't want a lot of phone calls from users. They aren't set up to handle the volume of calls and building a complete customer service call center could render their entire operation unprofitable. While it's inconvenient to the user, they'd rather handle the problems through email, which is far more cost effective.

The designers at Amazon have deliberately made the process of calling them very unintuitive to encourage customers to find another way to resolve their problems. (We're not saying this is the right thing for Amazon to do, but their choice does have some sound logic behind it.)

Understanding How Intuitive Works

Once you understand how 'intuitive' works--what makes someone perceive a design to be intuitive--it becomes easier to make the decision as to whether an intuitive design is worth the extra effort. The knowledge your users have when they arrive at the design (current knowledge), what knowledge they'll need to complete their tasks (target knowledge), and what the design needs to do to help them complete the task (the gap) are the key ingredients for making an interface that seems 'intuitive' to your users.
Lecture 7: Creative mornings with Ben Chestnut [video transcript];

http://blog.chrisbarber.co/transcript-creative-mornings-with-ben-chestnut

Ben Chestnut

Ben: Georgia Tech rejected me. I was like, "Screw you, guys. I'm just going to go to UGA." I always sound like A-hole when I say this but if you're working for me, you're a creative person, it's not my job to make you happy. If you search for the word boredom in our app, the whole screen turns into an asteroids game.

This money is like a pain in the ass to count. Are you guys having problem?

Speaker 2: Today, we have Ben Chestnut and he is the co-founder of MailChimp, a little company here in Atlanta and I'm going to read his bio real quick. He started college as a Physics major at UGA. He discovered Industrial Design. Then, he transferred to Georgia Tech and he got a degree in Industrial Design. He started technical Rocket Science Group, a web development agency in 2001. MailChimp was a side project of the Rocket Science Group to help their client send email. It eventually became their sole focus and now, they have 1.2 million users worldwide.

Without further advocacy, let's give it up to Ben Chestnut.

Ben: This is exciting for me because for the last 10 years of my life, pretty much I only got invited to speak about email marketing so finally, I get to talk about something else. I love email but I have nothing more to say about email. It's like I've been talking about this like if you have something interesting to say, send an email. Otherwise, don't send anything. It's really that simple.

I actually tell employees spread rumors in the industry that I'm a germaphobe and I won't travel and that way, people would stop asking me to talk about email marketing. Ten years, I've had all this thoughts and they've been stuck up here because no one has asked me to talk about them. Then Tina invited me. She's, "You can talk about anything you want." This is like just to warn you. This is 10 years of stuff that have been tormenting me. I hope I can do this here.

What I want to talk about is controlled chaos and the maximization of the entropic states as applied to steam engines and creative environments. An insanely laborious title and probably the most uncreative title ever for Creative Mornings but running a creative environment and a creative office is really hard work. It's unbelievable hard work and that's kind of what I want to convey in this. This is going to be really chaotic lecture by the way just to set your expectation.

As we got closer to you today, I sent an email to Tina kind of at point A to point B, how did I get to MailChimp. I'll give you a little background. I grew up in a creative household. My brother was a painter and also a musician. He played the guitar and he fiddled with electronics and my sisters were into graphic arts.

They were always making collages and stuff. My mother was an aspiring chef. My father was an aspiring writer but also a computer programmer. We were just always making stuff at home and I was into cartooning. I thought I wanted to be a cartoonist when I grew up. I would take ... I would steal post-it notes from my sisters and my brother and make little cartoons and I'll take them to school and show them off, flip them on the bus and the kids would gather around like, "Oh that so cool."

Then I started to sell them. I turned it into a business. I would stay up late at night and I should have been doing my math homework but I would make this cartoons and sell them for 50 cents which is so stupid I ... all night and I ... 50 cents. Really bad at math.

I would take this to school and they love them. They're like little stick figures that run across the screen, bouncing balls and people loved it and they wanted more so I would make a car come in and run over the people. They're like, "Man, that's awesome. Do more, more, more." I had to make jets come in and drop bombs on the cars and the cars would explode and little tires would bounce around on the screen and they wanted more, more.

I found myself I was 10 or something and I was already dealing with A-hole clients, really demanding. I was just 10 and I couldn't wake up in the morning. I didn't want to get on the bus anymore. I got out of it. I just totally got out of the business. My dad, he bought us an AMStrad PC and I guess I wanted him to be a computer programmer. He said, "You know, want to learn this stuff?" I was like, "No. No. Numbers, no. No." He didn't push me. He said, "That's fine." I didn't touch it for a while and went to RadioShack 1 day and he takes this box off the shelf. He's like, "What do you think, huh? Huh?" I think it was called Paint Deluxe Pro. I tried to go back and find this. It looks like one of the first programs from Electronic Arts which is like a gaming company now but it's Paint Deluxe but all I remember is on the cover was this giant tiger. It was a huge tiger face and I was like, "Tiger." I was like, "I want to learn that."

He bought it. It was hundreds of dollars and he never spent that much money. He's just, "Let's get it." It was 5 floppy disks, you had to load this like the 5 inch kind like and there was no hard drive. It's in the RAM. You have to load this in order to start the program. I totally fell in love with this. I was drawing on the computer all the time and I knew that's what I wanted to do with the rest of my life, just draw things on computers.

There was no name for this profession. It was new. There were no [inaudible 00:06:22] commercials like computer graphics or anything. I had no idea so I just figured that I had to be an engineer like engineers got to play with computers and draw stuff. I went all through high school taking drafting classes just thinking, "I got to be an engineer so I'll take drafting."

Then, it was time to apply to college and Georgia Tech rejected me.
The School of Mechanical Engineering was like, "No. Your Math is really bad." I was like, "Screw you, guys. I'm just going to UGA and I'm going to study Physics." Two years, I studied Physics and it was really... I looked like him. I was very depressed. It was like, "All right. I know how the world works now. What I'm going to do with that?" My sister, she was working in a creative company. She was at Hallmark actually designing cards. Her friends heard about me and sent me a care package in the mail and I remember getting this. It was a college catalogue for art center in Pasadena. You guys know about this school out there? I got in and I was like, "Wow!" I learned about this profession of Industrial Design. I was like, "Man, this is awesome." The suits and... He [inaudible 00:07:45] he's always got like a cigarette with his pictures. These guys get to play with computers. They draw stuff and there's no math. Not that much so I was like, "I want to do that now." Pasadena is extremely expensive that school so my dad was like, "No. No. No. No."

It turns out, yeah. Georgia Tech has a school of Industrial Design and I remember driving from Athens to Atlanta to talk to the director. It was Bill Bullock at the time. I don't know if anybody knows him but he looked at my portfolio. I drew 2 pictures and he said, "You know, you've got potential. It's obvious you haven't been doing this so if you were to apply today, we would say no but if you apply to the Georgia Tech School of Physics..." like nobody wants to learn Physics, "Go to Physics, they'll let you in and then you just transfer." He's like, "If you transfer, no one looks at your portfolio. You can just get in, you know."

I was like, "Really?" These designers, they're really sneaky people. I applied and they actually let me in. I was like, "Suckers." I got in and I was really, really in the product design. I love product design. I would probably be designing cars today but I suck with X-ACTO knives. These are the tool of the devil. They're cylindrical. You guys use these things? They spin in your hand. They're not right. I said, "Screw this. I'm going to get into Web Design." It's like... and... With Web Design it's like clean, you don't have to sand anything or cut anything. It's just like pixels. It's nice clean computers.

Then so it's like 1 thing led to another and I ended up at MailChimp. Yeah. Now, I'm like the CEO of a software company and like that's... I'm the happiest I've ever been. It's the most challenging and stressful I've ever been but I'm the happiest and I didn't start off looking to be the CEO of a company. I wanted to be a cartoonist and I think that's my first lesson is you always hear people say, "Do what you love. Do what you love." It's partly true but if it's business, if you start a business doing what you love, it will kill you. It will kill your passion.

If you like to bake and you start a bakery, you will have baking very soon. I love what you do better because it's wherever you're at just be good at it, embrace it, love it, and eventually success will find you. I actually believe that but you don't forget your passion. You never forget it. After all these years, we actually made a coloring book. We're a software company and we made this coloring book and it's called love what you do and it's all about our little mascot, Freddie, doing little things in life but really loving it just finding joy.

I just love this message and we printed out thousands of these things and we just send them out to customers randomly and they send back pictures of their kids coloring at the laundry mat and they're all bored and it's just... very touching for me. This is what I also want to do anyway, mission accomplished.

It got featured on this blog called Swiss Miss, some little design thing and it... I don't know, Tina, if you notice but the editor of [inaudible 00:11:17] Company apparently loves Swiss Miss and she made a writer contact me and do a story on us. That was neat. It was all about creative cultures and how we give you permission to be creative in the company and they talked about the whacky things that go on in the company and I'm like I don't know if you guys use MailChimp or but we had this mascot Freddie which just does random things while you're building your newsletter like, "Why I'm smiling. I'm not wearing any pants?"

People love it and he says lots of other random stuff that our customers have sent us and we started to get some designers saying, "You know, I love this but my clients, you know, he's got a stick up his butt. Can you turn this off in any way? Is there a button?" We made a button. Let's see. Yeah. It's called "Party Pooper Mode." You just activate that and the monkey goes away. They love it and they snap tweak picks of it. It's like a secret with us in the designers.

There are all kinds of little touches in the app like when you hit the send button, just underneath it, we say this is your moment of glory. That's really... I can't take credit for any of this stuff. All I can say is I tried to make an environment where people on the team can just play around this. I discovered this stuff along with our customers. I will send emails like, "Who did this? This is cool." You'll see people tweet all about this. They just love these tiny touches.

If you search... I hope you can see this. If you search for... do an Easter egg. It's this crazy like Easter eggs in the app and we don't say like I don't say, "Go do an Easter egg." I think I ask the guys like the CSS is way too bloated. It's 500 K. Knock it down. The guy started researching efficient coding methods and stuff and somewhere in the process, they found this game maybe as an exercise in efficient coding or something like that so they shaved down hundreds and hundreds of K and then they added this game back in because they had room. Snuck it back in.

That's where creativity comes from. We used to spend a lot of money on Google Ad words and we still do in case Google is watching. We still spend money but we launched a premium program a little while ago and that's... It really, really ramped up our user base and we could save a lot money from Google and we looked at the money that we were spending over there. We said, "You know, we could pocket it or let's maybe invest in our customers."

We started making these T-shirts and we'll ship them all over the world and it's like a nice surprise when customers like graduate from free to paid. You win a T-shirt and we'll ship it to you and we're getting photos back from all over like this fancy places that I hope to be able to travel to one day.
We're starting to get sightings like, "Hey, I saw you on TV." We'll get stuff like this. This is some guy on Hawaiian TV, I think. This is neat. Then like this is a print ad for some gym apparently and like that guy is wearing a Freddie T-shirt. It's kind of cool and that's MC Hammer. He's not wearing the shirt but like over there, that guy is. Close enough. I love that. We make these knitted hats and we just shipped this out randomly as well and our customers post pictures of this. They love this stuff and naturally, they started putting them on dogs and we would retweet this and then, of course, the cat people wanted something so we had to start making cat hats.

It's actually like a friend of a friend of a family member in Thailand that knits this like on one of the street markets and that was like ... I think she's probably super rich out there because we buy thousands of these things every month and like ... I remember writing the email like people were demanding cat hats so ... It's like, "This is going to sound weird but can you make this for cats?" They wrote back like in 5 minutes like, "Yes. Send us the measurements for an American cat." They're not obese. We made a bunch of this and we sent them out and we get ... Cats don't like hats. That's one freaking out. That gave us an idea like all of this like randomness, it starts giving us idea is we made an iPhone app called "Pyow!" Yes, actually, thank you. We're into cats for a little while and we're like, "Hey, let's make Pyow and we'll make like laser shoot out of his eyes because this is like a red laser app, it scans QR codes for our customers who wanted like send coupons and stuff.

Of course, this was like, "Wow! That's kind of neat." We need to make like cat shaped hats as well. In our office, I love like if you say, "Can you hand me a cat hat?" "Which one like the monkey shaped ones for cats or the cat shaped ones for dogs and humans?" You have to be specific when you say cat hats at MailChimp. People see stuff like this in our company and they see articles and stuff and the ask me like, "What's the formula? What's the formula for running a creative company?" I was like, "Is it that simple?" I never know the answer. It's such a weird question. You're like, "How do you have a creative company," because I think companies are legal entities. They're not creative. They're just pieces of paper. It's these people. You're totally missing then point. People want to be creative.

I wanted to share what I learned about humans. This is while I was designing refrigerators in Iowa. While I was a Industrial Design student, I actually interned, I think it was '96 like whereever ... When the Olympics were in Atlanta, I was in Iowa, like I'm never where the action is. I was out in the cornfields designing refrigerators in this in-house design studio and the designers were awesome people and the product managers that they were work with were awesome people. I learned a lot about business and managing business and focus groups and all that stuff.

When the 2 groups had to get together, it was pure hell. It was ... The passive-aggressive tension in the room was just crazy and I was just like a stupid intern but even I could tell, "Man, these guys hate each other." I never knew why but years later, I'm piecing things together and I understand now. They would secretly buy these Japanese and Korean refrigerators and ship them in and dissect them and look at them and they'd say, "Ooh, ahh. Look, it's got these floral prints on them. they've got curves ..." and this is '96 so like curves were new. They're like, "Wow, curves." They didn't beep. They sang like the birds would chirp and they would sing music and we're like, "Man, we want to do that." The whole company was tooled. The factories were tool to make white sheet metal boxes. You could white, glossy white, off-white, textured white, or cream. That's all we could do. It was incredibly frustrating to get to the point where they could do something beautiful. They're doing beautiful stuff now but it would be like 3 or 4 years. The designers and the managers, they all hated each other because they just couldn't get anything creative done. Really, really frustrating.

My takeaway was humans really want to create lots of cool stuff and they want to see other people using cool stuff. That's all they want in life. If you can create a business that takes advantage of this, you might have a creative company so to speak. The thing is you have to set up a business to take advantage of this and most businesses they're set up in a weird way. They make a fundamental mistake somewhere along the way. I thought I'd explain it. This is part of the 10 years of pent-up frustration about business. Here's an example. An entrepreneur has an idea usually. He wants to start a company. A business is like the steam machine like you don't know how this works. You start up a business. You're like, "Oh, if I tweak this knob, I think money comes out." If I adjust the screw or like maybe make the pulley or something tighter like more money will come out. That's the first couple of years and then after a little while, you're like, "Wait a minute. Wait a minute. Wait a minute." Two knobs. What happens then like, "Holy shit. Wow. ..." Then, you're like, "Wow, man. I kind of get this stuff. This is kind of cool." Now, I'm going to start thinking big like Richard Branson big. You start learning about key performance indicators like my KPIs are all like knobs. I'm like him but knobs on top of knobs and I've got knobs down here and then like bam. That's usually what happens and this is where things begin to shake with the company like, "All right. This money is like a pain in the ass to count." Do you guys have that problem like it's ... Pain in the ass, right? It's like everywhere. You need a manager to help you organize and stack this stuff because that's what managers do. They organize and they create order. Managers are good. I'm a manager. You need managers. They create order. You need that. The thing is ... This is not where things go wrong. Things go wrong when that original entrepreneur, the creative guy, says, "You know what? I deserve a break. I'm going to delegate now. The business is running itself. I can sort of like step back a little bit. Hands off." That's where things go wrong. They're like, "I'm going to take up like extreme sex surfing or something, you know," I don't know. I don't know what that is but I want to do that. He's out there like living it up. That's what business people
Nature, it's just a part of nature, chaos. You need to get used to it but the thing is managers hate disorder. They don't like this entropy stuff. It's inefficient. It ruins their sorting. If they had their way, S, entropy would be 0 and I believe the way that the equations work out is without S, you don't get Q and Q is part of the equation for work and... Basically, no chaos, no work, not output. They have their way. There'll be no pigsties which means no pigs. No pigs, no bacon. No bacon, no Baco Bits. We need chaos.

Chaos is good. You have to embrace chaos. I think my job as a manager of a creative kind of company and creative people is to find ways to create chaos. Little controlled chaos, not like, I don't slam employees with chairs. Nothing like, "Ooh yeah." Nothing like that.

I want to talk about the little ways that I try to create chaos in the company. This is a really big idea and I didn't have time to put text on this slide. It's ironic. This is like I think innovation and creativity comes from just assembling pieces from other stuff in weird ways. You're like... I try to tell people, "Don't worry about big ideas. Just keep making the stuff." Build little things. Build prototypes. Sketch this. You want to learn a new programming language, go ahead but don't take a 2-year course. Just learn a little bit and make something. You got 2 weeks to do it. Two weeks is the ideal timeline at least for me.

After 2 weeks, I don't want to hear you talking about it anymore. You keep it fast-paced and you're just making junk. It feels like just parts and that's what I tell people all the time like put it in the parts bin. You might launch it. It might have nothing to do with email marketing, nothing to do with MailChimp. Doesn't help us one bit with the business but just save it because we will use it one day.

Then, you want to avoid meetings. You want to let people stay and work on their stuff and you need meetings every once in a while but you keep it to a minimum so people can work on stuff but then, I always call myself like a little bumblebee like I buzz around from desk to desk and I ask people like, "What are you working on? What are you working on? What are you working on? What are you working on?"

I never praise people like, "Oh, that's cool," or anything. I just say, "What are you working on? Okay. What are you working on?" I just remember it because I think my job is to go around and say, "Oh, you're working on this but you need a logo like, oh, like Erin over there designed a logo and he doesn't have an app to give it to so like you guys should connect." This is hard work. I could just say, "You know, delegate. You guys go have a meeting and like focus on 2 projects." This is much more hard work but you just have to deal with it. You don't delegate the creativity away. You deal with it. It is difficult, time consuming.

I don't feel like I'm doing my job if I'm not buzzing around like a bumblebee. If you're lucky, you can put together these pieces in unique ways. You guys have probably seen this poster. You find creative ways of assembling these pieces and like if you have good managers, they'll take it from this level to something that you can sell. We have a guy in our company called Neo. He just loves it when it's time... when the creatives like put together their stuff, he comes in and
he’s like, “All right. Are you ready to make money out of this stuff?” He’ll turn into something like this.

One of the reasons I was so stressed is I wasted a month. I didn’t want to come up here and describe myself as a bumblebee. It felt unmanly like flowers. I spent a month ordering guerrilla warfare books and art of war and I was like, "Try to learn about this tactics," like I thought that wouldn’t be the topic or the theme. It didn’t work. It was ... I think the FBI is just watching me now because I bought all these books. I just went back to bumblebees and for a minute, I was like maybe like the Transformer Bumblebee. He’s awesome but I give up. I’m just not ... That’s me. That’s me. I’m a bumblebee. I’m going around. I’m connecting these random chaotic ideas right, and you’re just keeping it fast-paced, make people keep making things. I wanted to show you how this happens at our company. This is ... He’s a programmer, Jesse. He doesn’t shave. He lets his hair grow until he’s done with the project and then he shaves so we know when he’s working on something and when he’s done with something. He grows and then, he was ready to shave and our video guy, Josh, he’s like, "Can we film you like when you shave?" He’s like, "Whatever." It’s kind of random silly stuff and so he films it and they thread it backwards. Nothing. No rocket science there but whatever. We post it at the [inaudible 00:31:11] or something and customers got a chuckle and no big deal.

People think this is like creative culture. No. That’s not ... That’s a piece. This is like a part here. This is work because later on, our creative director starts thinking like, "Whoa. You know, he kind of looks like this Viking terminator like robot thing, right?" He starts like getting into Vikings all of a sudden. Ron was talking about Vikings and I was like, "Shut up about Vikings." He was like, "We got to make like an app called Enforcerator, but it’s ... Dude it says, Enforcerator. He spelled it right. Eventually, he got it right. He’s sketching Vikings. He’s just obsessed with Vikings for a while.

It really goes nowhere but he’s sketching swords and skulls and stuff and it turns out Chad, our lead engineer upstairs, is working on something called alter ego and it’s a 2-factor security app. He doesn’t have a logo. He’s like, "Maybe the design geniuses can come up with something," and he’s like, "Hey, we have a sword." We got this done. The whole project was done in about 2 weeks. We launched this thing and what’s really beautiful, the whole human thing. You want to build something cool and you want to see other people use it. We built this in 2 weeks and made it free for our 1.2 million users. We get to see it in action.

Version 1 that we knocked out in 2 weeks, it didn’t ... It was a mobile web app, not a native iPhone or Android app. You have to log in to your browser on your phone. It was okay. We just want to get it live though. It turns out we have ... We have a mobile lab 2 doors down from Chad and I told him about this. I don’t say, "Hey, we’re working on something. Can you help?" I’d like to say, "We just launched something. Can you help?" I think that’s important. We just launched something now. Can you go back and help us build a native app and they were actually wanting to tinker with Android apps. They were into iPhone but they wanted to ...
money than us." There was no leap or anything. It's like while you're in there doing your work hustling, that pops up. Yeah. It's an excellent question. It's like ... If I can repeat it, it's, "How do you get back to the numbers when you're working on happiness all the time? Can you really turn that into money?" The part about that's hard for me to convey, I tried it and I took out the slides. It's, "I don't care about your happiness." A lot of people worry about that like they think that I'm here doing this fun stuff for happiness and I always sound like an A-hole when I say this. If you're working for me, you're creative person, it's not my job to make you happy. We have a business. It's hard work. We're making money but I'm going to give you opportunities to be creative by keeping it fast-paced and making you do random stuff. You're going to be creative and if you're a creative person, then you'll be happy indirectly.

I'm not ever focused on happiness. I don't ... You are a robot. I plug you in and you make me money. No. That's ... Tiger to chimp. That's a good question. I don't have any idea. I know we got chimp from ... We had to pull an all-nighter and it was during the Super Bowl and like we really resented that working for a client and there was no YouTube like people were posting to some advertising site all the commercials. We would work and then pull up the site and watch Super Bowl commercials and they were all chimps. We were like, "People like chimps.

Speaker 4: Earlier, you were talking about love what you do versus doing what you love. [Inaudible 00:38:58] and maybe your last answer about not caring my happiness [inaudible 00:39:01] but is it your contention that for a lot of creatives, you do start a business based on something that they love that they should go out and find something to do that would not necessarily be what they love but is able to finance what it is that they love. Does that make sense?

Ben: Yeah.

Speaker 4: For instance, if you start a business as a bakery and you end up hating baking. Now, you're like, "Okay. I hate baking." This is something that my wife is doing [inaudible 00:39:31] photographer. It's like all the pains of [inaudible 00:39:33] a photography business is making her hate photography. I'm like, "Sweetheart, you know, you should just go back to you doing what you love." What's your take on that, Ben? Is it better to then in the real world find something that can finance what you love whether or not it's a business that you love doing and then spend your free time doing what you love? Does that make sense?

Ben: Yeah, it does. It's hard to answer. One thing you can just get a manager and to help with a lot of that stuff that you hate so much, managers are really good for that kind of stuff and they love that actually. They're doing what they love. Yeah. It's a really hard one to answer. I feel like I never got to do exactly what I love but I still try to be really good and I ended up loving whatever ... wherever ... I had a stint as a banner ad designer for 2 years. I loved it. I was pretty good at it. Eventually, all of these things add up to some kind of knowledge where in the end, you will be able to go back and do your passion but while you're working on business, business really isn't about that passion.

There's a book that like weird businesses sales e-book but it's called the e-myth. I don't know if anybody's read this but it talks a lot about this sort of thing where, it's your passion, you start a business and it becomes a chore and you hate it. They're ways around it. I'd say just be prepared to lose the passion for a few years but keep it. It will come back later. You might end up with a business that does email and not cartoony but you still get your passion done.

Would I have done MailChimp sooner? Yeah. I sometimes think about that but the way that the economy work out when we started it, it was right after 9/11 and there was ... like a real estate fallout before that and right before that was like the dot com bust and we call ourselves cockroaches like no matter what happens, we can survive. That kind of pain really helped us get stronger. I don't know if I would go back and change that. I see lot people now like, "Man, they start up in good times and like their business jus ramps up," and when you see them in hard times, it's really, really rough for them. We're like, "Pssh. That's every day." Cockroaches. Thank you, all.

Speaker 2: Thanks.
Lecture 7: What Makes Marriage Work?

http://www.psychologytoday.com/articles/200910/what-makes-marriage-work

It's how you resolve conflict that matters most.
By John Gottman, Nan Silver

Research indicates you can get mad as hell or avoid conflict altogether. But the positivity must outweigh the negativity by five to one.

If you are worried about the future of your marriage or relationship, you have plenty of company. There's no denying that this is a frightening time for couples. More than half of all first marriages end in divorce; 60 percent of second marriages fail. What makes the numbers even more disturbing is that no one seems to understand why our marriages have become so fragile.

In pursuit of the truth about what tears a marriage apart or binds it together, I have found that much of the conventional wisdom—even among marital therapists—is either misguided or dead wrong. For example, some marital patterns that even professionals often take as a sign of a problem—such as having intense fights or avoiding conflict altogether—I have found can signify highly successful adjustments that will keep a couple together. Fighting, when it airs grievances and complaints, can be one of the healthiest things a couple can do for their relationship.

If there's one lesson I've learned in my years of research into marital relationships—having interviewed and studied more than 200 couples over 20 years—it is that a lasting marriage results from a couple's ability to resolve the conflicts that are inevitable in any relationship. Many couples tend to equate a low level of conflict with happiness and believe the claim "we never fight" is a sign of marital health. But I believe we grow in our relationships by reconciling our differences. That's how we become more loving people and truly experience the fruits of marriage.

Although there are other dimensions that are telling about a union, the intensity of argument seems to bring out a marriage's true colors. To classify a marriage, in my lab at the University of Washington in Seattle, I look at the frequency of fights, the facial expressions and physiological responses (such as pulse rate and amount of sweating) of both partners during their confrontations, as well as what they say to each other and in what tone of voice they interact verbally.

But there's much more to a successful relationship than knowing how to fight well. Not all stable couples resolve conflicts in the same way, nor do they mean the same thing by "resolving" their conflict. In fact, I have found that there are three different styles of problem solving into which healthy marriages tend to settle:

- Validating. Couples compromise often and calmly work out their problems to mutual satisfaction as they arise.
- Volatile. Conflict erupts often, resulting in passionate disputes.
- Conflict-avoiding. Couples agree to disagree, rarely confronting their differences head-on.

Previously, many psychologists might have considered conflict-avoiding and volatile marriages to be destructive. But my research suggests that all three styles are equally stable and bode equally well for the marriage's future.

"HEALTHY" MARRIAGE STYLES

One of the first things to go in a marriage is politeness. As laughter and validation disappear, criticism and pain well up. Your attempts to get communication back on track seem useless, and partners become lost in hostile and negative thoughts and feelings. Yet here's the surprise: There are couples whose fights are as deafening as thunder yet who have long-lasting, happy relationships.

The following three newly married couples accurately illustrate the three distinct styles of marriage.

Bert and Betty, both 30, both came from families that weren't very communicative, and they were determined to make communication a priority in their relationship. Although they squabbled occasionally, they usually addressed their differences before their anger boiled over. Rather than engaging in shouting matches, they dealt with their disagreements by having "conferences" in which each aired his or her perspective. Usually, they were able to arrive at a compromise.

Max 40, and Anita, 25, admitted that they quarreled far more than the average couple. They also tended to interrupt each other and defend their own point of view rather than listen to what their partner was expressing. Eventually, however, they would reach some sort of accord. Despite their frequent tension, however, they seemed to take much delight in each other.

Joe, 29, and Sheila, 27, said they thought alike about almost everything and felt "an instant comfort" from the start. Although they spent a good deal of time apart, they still enjoyed each other's company and fought very rarely. When tension did arise, both considered solo jogging more helpful in soothing the waters than talking things out or arguing.

Not surprisingly, Bert and Betty were still happily married four years after I'd first interviewed them. However, so were Max and Anita, as well as Joe and Sheila. Marriages like Bert and Betty's, though, which emphasize communication and compromise, have long been held up as the ideal. Even when discussing a hot topic, they display a lot of ease and calm, and have a keen ability to listen to and understand each other's emotions.
That's why I call such couples "validators": In the midst of disagreement they still let their partners know that they consider his or her emotions valid, even if they don't agree with them. This expression of mutual respect tends to limit the number of arguments couples need to have.

Anita and Max take a different approach to squabbling than do Bert and Betty, yet their marriage remained just as solid over time. How can people who seem to thrive on skirmishes live happily together? The truth is that not every couple who fights this frequently has a stable marriage. But we call those who do "volatile." Such couples fight on a grand scale and have an even grander time making up.

More than the other types, volatile couples see themselves as equals. They are independent sorts who believe that marriage should emphasize and strengthen their individuality. Indeed, they are very open with each other about their feelings--both positive and negative. These marriages tend to be passionate and exciting, as if the marital punch has been spiked with danger.

Moving from a volatile to an avoidant style of marriage, like Joe and Sheila's, is like leaving the tumult of a hurricane for the placid waters of a summer lake. Not much seems to happen in this type of marriage. A more accurate name for them is "conflict minimizers," because they make light of their differences rather than resolving them. This type of successful coupling flies in the face of conventional wisdom that links marital stability to skillful "talking things out."

It may well be that these different types of couples could glean a lot from each other's approach--for example, the volatile couple learning to ignore some conflicts and the avoidant one learning how to compromise. But the prognosis for these three types of marriage is quite positive--they are each healthy adaptations to living intimately with another human being.

THE ECOLOGY OF MARRIAGE

The balance between negativity and positivity appears to be the key dynamic in what amounts to the emotional ecology of every marriage. There seems to be some kind of thermostat operating in healthy marriages that regulates this balance. For example, when partners get contemptuous, they correct it with lots of positivity--not necessarily right away, but sometime soon.

What really separates contented couples from those in deep marital misery is a healthy balance between their positive and negative feelings and actions toward each other.

Volatile couples, for example, stick together by balancing their frequent arguments with a lot of love and passion. But by balance I do not mean a 50-50 equilibrium. As part of my research I carefully charted the amount of time couples spent fighting versus interacting positively--touching, smiling, paying compliments, laughing, etc. Across the board I found there was a very specific ratio that exists between the amount of positivity and negativity in a stable marriage, whether it is marked by validation, volatility, or conflict avoidance.

That magic ratio is 5 to 1. As long as there is five times as much positive feeling and interaction between husband and wife as there is negative, the marriage was likely to be stable over time. In contrast, those couples who were heading for divorce were doing far too little on the positive side to compensate for the growing negativity between them.

WARNING SIGNS: THE FOUR HORSEMEN

If you are in the middle of a troubled marriage, it can seem that your predicament is nearly impossible to sort out. But in fact unhappy marriages do resemble each other in one overriding way: they followed the same, specific, downward spiral before coming to a sad end.

Being able to predict what emotions and reactions lead a couple into trouble is crucial to improving a marriage's chances. By pinpointing how marriages destabilize, I believe couples will be able to find their way back to the happiness they felt when their marital adventure began.

The first cascade a couple hits as they tumble down the marital rapids is comprised of the "Four Horsemen"--four disastrous ways of interacting that sabotage your attempts to communicate with your partner. As these behaviors become more and more entrenched, husband and wife focus increasingly on the escalating sense of negativity and tension in their marriage. Eventually they become deaf to each other's efforts at peacemaking. As each new horseman arrives, he paves the way for the next, each insidiously overriding a marriage that started out full of promise.

THE FIRST HORSEMAN: CRITICISM

When Eric and Pamela married fresh out of college, it soon became clear that they had different notions of what frugality meant. Pamela found herself complaining about Eric's spending habits, yet as time passed she found that her comments did not lead to any change on her husband's part. Rather, something potentially damaging to their marriage soon began occurring: instead of complaining about his actions, she began to criticize him.

On the surface, there may not seem to be much difference between complaining and criticizing. But criticizing involves attacking someone's personality or character rather than a specific behavior, usually with blame. When Pamela said things like "You always think about yourself," she assaulted Eric, not just his actions, and blamed him for being selfish.

Since few couples can completely avoid criticizing each other now and then, the first horseman often takes up long-term residence even in relatively healthy marriages. One
reason is that criticizing is just a short hop beyond complaining, which is actually one of the healthiest activities that can occur in a marriage. Expressing anger and disagreement makes the marriage stronger in the long run than suppressing the complaint.

The trouble begins when you feel that your complaints go unheeded and your spouse repeats the offending habits. Over time, it becomes more and more likely that your complaints will pick up steam. With each successive complaint you're likely to throw in your inventory of prior, unresolved grievances. Eventually you begin blaming your partner and being critical of his or her personality rather than of a specific deed.

One common type of criticism is to bring up a long list of complaints. I call this "kitchen sinking": you throw in every negative thing you can think of. Another form is to accuse your partner of betraying you, of being untrustworthy: "I trusted you to balance the checkbook and you let me down! Your recklessness amazes me." In contrast, complaints don't necessarily finger the spouse as a culprit; they are more a direct expression of one's own dissatisfaction with a particular situation.

Criticism also tends to be generalizations. A telltale sign that you've slipped from complaining to criticizing is if global phrases like "you never" or "you always" start punctuating your exchanges:

Complaint: "We don't go out as much as I'd like to.
Criticisms: "You never take me anywhere."

Being critical can begin innocently enough and is often the expression of pentup, unresolved anger. It may be one of those natural self-destruct mechanisms inherent in all relationships. Problems occur when criticism becomes so pervasive that it corrodes the marriage. When that happens it heralds the arrival of the next horseman that can drag you toward marital difficulty.

THE SECOND HORSEMAN: CONTEMPT

By their first anniversary, Eric and Pamela still hadn't resolved their financial differences. Unfortunately, their fights were becoming more frequent and personal. Pamela was feeling disgusted with Eric. In the heat of one particularly nasty argument, she found herself shrieking: "Why are you so irresponsible?" Fed up and insulted, Eric retorted, "Oh, shut up. You're just a cheapskate. I don't know how I ended up with you anyway." The second horseman--contempt--had entered the scene.

What separates contempt from criticism is the intention to insult and psychologically abuse your partner. With your words and body language, you're lobbing insults right into the heart of your partner's sense of self. Fueling these contemptuous actions are negative thoughts about the partner--he or she is stupid, incompetent, a fool. In direct or subtle fashion, that message gets across along with the criticism.

When this happened, they ceased being able to remember why they had fallen in love in the first place. As a consequence, they rarely complimented each other anymore or expressed mutual admiration or attraction. The focal point of their relationship became abusiveness.

What Pamela and Eric experienced is hardly uncommon. When contempt begins to overwhelm your relationship, you tend to forget your partner's positive qualities, at least while you're feeling upset. You can't remember a single positive quality or act. This immediate decay of admiration is an important reason why contempt ought to be banned from marital interactions.

Recognizing when you or your spouse is expressing contempt is fairly easy. Among the most common signs are:

- Insults and name-calling
- Hostile humor
- Mockery
- Body language--including sneering, rolling your eyes, curling your upper lip.

It is easy to feel overly critical at times, and it is human to state criticism in a contemptuous way now and then, even in the best relationships. Yet if abusiveness seems to be a problem in your relationship, the best way to neutralize it is to stop seeing arguments with your spouse as a way to retaliate or exhibit your superior moral stance. Rather, your relationship will improve if you approach your spouse with precise complaints rather than attacking your partner's personality or character.

THE THIRD HORSEMAN: DEFENSIVENESS

Once contempt entered their home, Eric and Pamela's marriage went from bad to worse. When either of them acted contemptuously, the other responded defensively, which just made matters worse. Now they both felt victimized by the other--and neither was willing to take responsibility for setting things right. In effect, they both constantly pleaded innocent.

The fact that defensiveness is an understandable reaction to feeling besieged is one reason it is so destructive--the "victim" doesn't see anything wrong with being defensive. But defensive phrases, and the attitude they express, tend to escalate a conflict rather than resolve anything. If you are being defensive, you are adding to your marital troubles. Familiarize yourself with the signs of defensiveness so you can recognize them for what they truly are:

- Denying Responsibility. No matter what your partner charges, you insist in no uncertain terms that you are not to blame.
- Making Excuses. You claim that external circumstances beyond your control forced you to act in a certain way.
- Disagreeing with Negative Mind-Reading. Sometimes your spouse will make assumptions about your private feelings, behavior, or motives (in phrases such as "You
think it's a waste of time" or "I know how you hate it"). When this "mind-reading" is delivered in a negative manner, it may trigger defensiveness in you.

- Cross-Complaining. You meet your partner's complaint (or criticism) with an immediate complaint of your own, totally ignoring what your partner has said.
- Repeating Yourself. Rather than attempting to understand the spouse's point of view, couples who specialize in this technique simply repeat their own position to each other again and again. Both think they are right and that trying to understand the other's perspective is a waste of time.

The first step toward breaking out of defensiveness is to no longer see your partner's words as an attack but as information that is being strongly expressed. Try to understand and empathize with your partner. This is admittedly hard to do when you feel under siege, but it is possible and its effects are miraculous. If you are genuinely open and receptive when your partner is expecting a defensive response, he or she is less likely to criticize you or react contemptuously when disagreements arise.

THE FOURTH HORSEMAN: STONEWALLING

Exhausted and overwhelmed by Pamela's attacks, Eric eventually stopped responding, even defensively, to her accusations. Their marriage went from being marred by poor communication to being virtually destroyed by none. Once Eric stopped listening to Pamela, their relationship became extraordinarily difficult to repair. Instead of arguing about specific issues, every confrontation degenerated into Pamela screaming at Eric that he was shutting her out: "You never say anything. You just sit there. It's like talking to a brick wall."

Stonewalling often happens while a couple is in the process of talking things out. The stonewaller just removes himself by turning into a stone wall. Usually someone who is listening reacts to what the speaker is saying, looks at the speaker, and says things like "Uh huh" or "Hmmm" to indicate he is tracking. But the stonewaller abandons these messages, replacing them with stony silence. Stonewallers do not seem to realize that it is a very powerful act: It conveys disapproval, icy distance, and smugness. It is very upsetting to speak to a stonewalling listener. This is especially true when a man stonewalls a woman. Most men don't get physiologically aroused when their wives stonewall them, but wives' heart rates go up dramatically when their husbands stonewall them.

The fourth horsemen need not mark the end of a relationship. But if your interactions have deteriorated to this extent you are at great risk of catapulting even farther down the marital cascade--becoming so overwhelmed by the negativity in your relationship that you end up divorced, separated, or living lonely, parallel lives in the same home. Once the fourth horsemen becomes a regular resident, it takes a good deal of hard work and soul-searching to save the marriage.

The four horsemen are not the end of the line. It is only after they turn a relationship sour that the ultimate danger arises: Partners seize on powerful thoughts and beliefs about their spouse that cement their negativity. Only if these inner thoughts go unchallenged are you likely to topple down the final marital cascade, one that leads to distance and isolation. However, if you learn to recognize what is happening to your once-happy marriage, you can still develop the tools you need to regain control of it.

Excerpted from Why Marriages Succeed or Fail by John Gottman, Ph.D., with Nan Silver. Copyright (C) 1994 by John Gottman. Reprinted by permission of Simon & Schuster.

KEYS TO IMPROVING YOUR MARRIAGE

When you're feeling overwhelmed, make a deliberate effort to calm yourself. This strategy eases the need to be defensive and to stonewall--two of the "Four Horsemen"--and undercuts the physical feelings that sustain the thoughts that maintain distress.

From the data gathered in our lab we've seen how quickly discussions fall apart as soon as one spouse's heart rate begins to soar. Learning how to calm down helps prevent unproductive fighting or running away from the important discussions you may need to have.

In addition, listening or speaking without being defensive helps to counter several destructive habits. If you happen to be a nondefensive listener, chances are it will make the cycle of negativity much less likely. And a nondefensive attitude on your part also helps to defuse the need to stonewall, particularly for men. But keep in mind that defensiveness is a two-way street; if you start speaking nondefensively, you will lessen your partner's need to be defensive.

Letting your spouse know that you understand him or her is also one of the most powerful tools for healing your relationship. It is an antidote to criticism, contempt, and defensiveness. Instead of attacking or ignoring your partner's point of view, you try to see the problem from his or her perspective and show that you think his or her viewpoint may have some validity.

When you've had one successful fight using these techniques, you may think you've fully mastered the strategies. But these lessons have to be practiced often. So often, in fact, that they become completely automatic. Each time you rehearse being nondefensive or validating is a new and different experience and it's important to keep trying even when you're tired and don't feel like it. The idea is that if you overlearn a communication skill, you'll have ready access to it when you need it most--during a heated argument with your spouse when you are physiologically aroused.

EXERCISE: HOW DO WE COMPARE?
Lecture 7: How to Build Products Users Love

This exercise gives you a chance to see the strengths of your marriage by comparing yourselves to other couples in your lives.
1. Each of you jot down the names of four different couples you both know. Two should be examples of "bad" marriages; two of "good" marriages.
2. Now share the names with one another and tell why you feel the good marriages work and the bad marriages don't. Perhaps you admire how one couple is raising their children, or you disapprove of the way another couple berates one another in front of company.
3. Talk about your own marriage in relation to these good and bad marriages. Compare the way you and your spouse manage to get through difficult times with the way each of these couples handle their challenges. Can you identify behaviors you want to avoid? Are there things you'd like to emulate?
4. Talk about your own ability as a couple to overcome hardship. Have you weathered episodes or incidents of which you're particularly proud? If so, how did you do it?

REPAIRING THE DAMAGE

Fortunately, in most relationships, there are ways of fixing things. I call these "repair mechanisms." Often, they are needed most when people are frustrated and angry, so they are said with some irritation or hurt, or even accompanied by an insult or threat. But they are repair mechanisms nonetheless.

Happily married couples use certain phrases and actions during an argument that prevent negativity from spiraling out of control. In effect, these conciliatory gestures act as a glue that helps to hold the marriage together during tense times.
1. Try to make comments about the communication process itself, such as "Please let me finish," or "We're getting off the topic," or "That hurt my feelings."
2. Comment on what's happening while it's taking place, not afterward.
3. Remind your partner that you admire and empathize with them despite the conflict.
4. Use phrases such as "Yes, I see," "Uh huh," or "Go on." These are little psychological strokes at which stable couples are masters.

Lecture 7: (BONUS) BJ Fogg’s Behavior Model

http://www.behaviormodel.org/

B.J. Fogg

What Causes Behavior Change?

My Behavior Model shows that three elements must converge at the same moment for a behavior to occur: Motivation, Ability, and Trigger. When a behavior does not occur, at least one of those three elements is missing.

Using my Behavior Model (FBM) as a guide, designers can identify what stops people from performing behaviors that designers seek. For example, if users are not performing a target behavior, such as rating hotels on a travel web site, the FBM helps designers see what psychological element is lacking.

The FBM also helps academics understand behavior change better. What was once a fuzzy mass of psychological theories now becomes organized and specific when viewed through my Behavior Model.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
The FBM highlights three principal elements, each of which has subcomponents. Specifically, the FBM outlines three Core Motivators (Motivation), six Simplicity Factors (Ability), and three type of Triggers. The subcomponents define the larger elements. For example, in the FBM the word Ability refers to the how the six Simplicity Factors work together in the context of a Trigger.

Many other people have proposed ways to understand persuasion and behavior change, dating back to Aristotle in ancient Greece. What makes my Behavior Model different from previous work? First, the FBM shows how behavior is the result of three specific elements coming together at one moment. Next, the FBM explains the subcomponents of each element. In addition, the FBM shows that motivation and ability can be traded off (e.g., if motivation is very high, ability an be low). Finally, the FBM applies most directly to practical issues of designing for behavior change using today's technology.

If you know of work related to the FBM, please share it on the page for references and connections. This will be an ongoing resource for people designing for and studying behavior change.

If you'd like to stay updated on my Behavior Model, sign up for the free newsletter. This way you will received new insights about FBM, as well as new ways to use these insights in industry and academic work.

The Fogg Behavior Model is part of a larger system that helps people design for behavior change. Shown in the graphic below, the larger context of the FBM will be the topic of an upcoming book for professionals in behavior change. Until then, see these resources:

- www.BehaviorWizard.org (see the 15 guides)
- www.BehaviorGrid.org
- captology.stanford.edu
- a short 2009 paper
- an updated 2010 paper (I think it's awful that Springer charge $25 for this, but don't get me started . . . )

Lecture 7: (BONUS): Customer Intimacy and Other Value Disciplines

http://hbr.org/1993/01/customer-intimacy-and-other-value-disciplines/ar/1

by Michael Treacy and Fred Wiersema

How was Dell Computer able to charge out of nowhere and outmaneuver Compaq and other leaders of the personal computer industry? Why are Home Depot's competitors losing market share to this fast-growing retailer of do-it-yourself supplies when they are all selling similar goods? How did Nike, a start-up company with no reputation behind it, manage to run past Adidas, a longtime solid performer in the sport-shoe market?

All three questions have the same three answers. First, Dell Computer, Home Depot, and Nike redefined value for customers in their respective markets. Second, they built powerful, cohesive business systems that could deliver more of that value than competitors. Third, by doing so they raised customers' expectations beyond the competition's reach. Put another way, these industry leaders changed what customers valued and how it was delivered, then boosted the level of value that customers expected.

The idea that companies succeed by selling value is not new. What is new is how customers define value in many markets. In the past, customers judged the value of a product or service on the basis of some combination of quality and price. Today's customers, by contrast, have an expanded concept of value that includes convenience of purchase, after-sale service, dependability, and so on. One might assume, then, that to compete today, companies would have to meet all these different customer expectations. This, however, is not the case.

Companies that have taken leadership positions in their industries in the last decade typically have done so by narrowing their business focus, not broadening it. They have focused on delivering superior customer value in line with one of three value disciplines—operational excellence, customer intimacy, or product leadership. They have become champions in one of these disciplines while meeting industry standards in the other two. (For a discussion of companies that excel at more than one discipline, see the insert "Masters of Two.")

Masters of Two

By operational excellence, we mean providing customers with reliable products or services at competitive prices and delivered with minimal difficulty or inconvenience. Dell, for instance, is a master of operational excellence. Customer intimacy, the second value discipline, means segmenting and targeting markets precisely and then tailoring offerings to match exactly the demands of those niches. Companies that excel in customer intimacy combine detailed customer
knowledge with operational flexibility so they can respond quickly to almost any need, from customizing a product to fulfilling special requests. As a consequence, these companies engender tremendous customer loyalty.

Home Depot, for example, is better than any other company in its market at getting the customer precisely the product or information he or she wants. And product leadership, the third discipline, means offering customers leading-edge products and services that consistently enhance the customer’s use or application of the product, thereby making rivals’ goods obsolete. Nike excels in product leadership in the sport-shoe category.

Companies that push the boundaries of one value discipline while meeting industry standards in the other two gain such a lead that competitors find it hard to catch up. This is largely because the leaders have aligned their entire operating model—that is, the company’s culture, business processes, management systems, and computer platforms—to serve one value discipline. Knowing what they want to provide to customers, they have figured out what they must do to follow through. And with the hard work of transforming their organizations behind them, they can concentrate on smaller adjustments that produce incremental value. Less focused companies must do far more than simply tweak existing processes to gain this advantage.

Companies that pursue the same value discipline have remarkable similarities, regardless of their industry. The business systems at Federal Express, American Airlines, and Wal-Mart, for example, are strikingly similar because they all pursue operational excellence. An employee could transfer from FedEx to Wal-Mart and, after getting oriented, feel right at home. Likewise, the systems, structures, and cultures of product leaders such as Johnson & Johnson in health care and pharmaceuticals and Nike in sport shoes look much like one another. But across two disciplines, the similarities end. Send people from Wal-Mart to Nike, and they would think they were on a different planet. Moreover, homogeneity exists only among leaders in the same value discipline; mediocre performers are not distinctive enough to look like anything except other mediocre performers in their own industries.

The conclusions we’ve drawn about the value disciplines are based on a three-year study of 40 companies that have redefined performance expectations in their markets. Through this research, we have come to understand what each value discipline demands of an organization and why.
8. Doing things that don’t scale
Lecture 8: Doing Things That Don't Scale, PR, and How to Get Started

http://startupclass.samaltman.com/courses/lec08/

Stanley Yang

Thanks for having me! I'm Stanley, the founder of DoorDash. It's really amazing to be here, because it wasn't actually that long ago when I sat in your seats. I was class of 2014, graduated in CS, as well as my cofounder Andy. For those of you who don't know what DoorDash is, we're building an on-demand delivery network for local cities. I want to start off with this photo that I took a few months ago. This was the night when we just raised our series A. I took this photo as I was walking back to where I lived; I actually lived in Roble at the time on campus. I took this photo because I realized just how ridiculous the combination of things I was holding in my hand was at the time. I was holding my CS247 homework, my tax forms (it was April – so I had to fill out my tax forms), that yellow speeding ticket, and right below that was a $15 million piece of paper I had just signed from Sequoia. And that kind of summarizes just how ridiculous our journey has been, doing this while I was at Stanford, and then transitioning this to an actual startup. I want to share with you that story today.

It all began two years ago in a macaroon store. It was my junior year at Stanford during the fall quarter. At the time, I was really passionate about building technology for small business owners. I sat down with Chloe, the owner of Chantal Guillon, a macaroon store in Palo Alto at the time, just interviewing her, trying to get feedback on this prototype we'd been working on, and also just learning about what her problems were in general. It was during this meeting when Chloe first brought up this problem of delivery. I remember she brought down this really really thick booklet. She showed me pages and pages of delivery orders, and a lot of these orders she had to turn down because there was no way she could have fulfilled them. She had no drivers, and she ended up having to personally deliver all these orders. That was a very interesting moment for us.

Over the course of the next few weeks, we talked to around 150 to 200 small business owners, and when we brought up this idea of delivery, they kept agreeing with us; they would say, "You know, we don't have delivery infrastructure. It's such a huge pain for us. There aren't any good solutions out there." This led us to wonder, delivery is such a common thing, such an obvious thing; why hasn't anyone solved this yet? Like, we must be missing something here right? We thought it was maybe because people had already tried this in the past, but they failed because there wasn't consumer demand for this. We asked ourselves, "How can we test this hypothesis?" We were just a bunch of college kids at the time. We didn't own trucks or delivery infrastructure or anything like that; we couldn't just build a delivery company overnight right? So how could we test this assumption we had?

We decided to create a simple experiment with restaurant delivery. We spent about an afternoon just putting together a quick landing page. When I went on the Internet, I found some PDF menus of restaurants in Palo Alto. We stuck it up there and added a phone number at the bottom, which was actually our personal cell phone number. And that was it. We put up the landing page and called it PaloAltoDelivery.com. This is actually what it looked liked (PowerPoint slide): It was super simple, ugly, and honestly we weren't really expecting anything - we just launched it. What we wanted to see was just would we receive phone calls, and if we got enough phone calls, then maybe this delivery idea was worth pursuing.

So we put it up there; we weren't really expecting anything, and all of a sudden we got a phone call. Someone called! They wanted to order Thai food. And we're like, "This is a real order; we're going to have to do something about it." So we're in our cars and we're like, "We're not doing anything right now,
might as well swing by, pick up some Pad Thai, and let's try to see how this whole delivery thing works." And we did. We delivered it to some guy up on Alpine Road I remember. We asked him, "How did you hear about us, what do you do?" He told us he was a scholar, and then he handed me his business card and told me he was the author of a book called We the People. That was our first ever delivery. It was like the best delivery/worst delivery you could ever ask for.

And then yeah, the next day we got two more phone calls. The day after that we got five, then it became seven, and then it became ten. And then soon we began to gain traction on campus through PaloAltoDelivery.com which is pretty crazy, because think about it: This was just a landing page. You had to look up PDF menus to place your orders and then call in. This isn't exactly the most professional-looking site, yet we kept getting phone calls; we kept getting orders. And that's kind of when we knew that we were onto something. We knew we found a need people wanted when people were willing to put up with all of this.

I think another key point to remember is we launched this in about an hour. We didn't have any drivers; we didn't have any algorithms; we didn't have a backend; we didn't spend six months building a fancy dispatch system – we didn't have any of that. We just launched because at the beginning it's all about testing the idea, trying to get this thing off the ground, and figuring out if this was something people even wanted. And it's okay to hack things together at the beginning.

At YC there's a mantra we like to talk about that is doing things that don't scale. So at the beginning we were the delivery drivers. We would go to class, and then after we would go deliver food. We were the customer support; you know I sometimes had to take phone calls during lectures. We spent afternoons just going down University Avenue just passing out flyers trying to promote DoorDash. We didn't have any dispatch system so what we had to do was use Square to charge all of our customers. We used a Google Doc to keep track of our orders. We used Apple's Find My Friends to keep track of where all of our drivers were. You know, just stuff like that, just hacking together solutions to try to get this thing off the ground. In fact at one point we were growing so fast that Square actually shut us down because we were under suspicion for money laundering. I mean think about it, we were getting small chunks of $15-$20 orders coming in at a rapid pace. Luckily, my cofounder Tony worked at Square so he just emailed some buddies there and everything was solved.

Another thing about doing things that don't scale is it also allows you to become an expert in your business, like driving helped us understand how the whole delivery process worked. We used that as an opportunity to talk to our customers, talk to restaurants. We did dispatching which helped us figure out - you know, we manually dispatched our drivers and that helped us figure out what our driver assignment algorithms should look like. We did customer support ourselves, getting real-time feedback from customers. I remember for the first few months when we got started, we would manually email every single new customer at the end of every night asking how their first delivery went, and how they heard about us. We would personalize all these emails: If I saw someone order chicken skewers from Orange Hummus, we would say "Oh I love Orange Hummus. How are your chicken skewers? How did you hear about us?" Feedback like that was really valuable, and customers really appreciated that.

I remember this one time - this was during YC - we had just come out of a meeting with one of our restaurant partners, and we heard about this ice cream place that had just opened up on University Avenue called Cream, and we wanted to go try it out. Then all of a sudden, our cofounder back at our office/house texted us saying "Oh we need drivers on the road; we got a huge spike in demand." So we debated for maybe about 10 seconds if we should go get ice cream or should we go deliver. We obviously went to deliver, but that kind of became our motivation on scaling, like you know, if we would scale, then we could go get ice cream next time.
Now of course we scale across different cities. Now we have to worry about building automated solutions, building dispatch systems, and figuring out how to match demand and supply - all that fancy technology stuff. But none of that mattered at the beginning because at the beginning it's all about getting the thing off the ground, and trying to find product-market fit.

Just to summarize, there are three things I would say I learned from doing DoorDash. First, test your hypothesis. You want to treat your startup ideas like experiments. The second thing is, launch fast. We launched in less than an hour with a really simple landing page. And finally, it's okay to do things that don't scale. Doing things that don't scale is one of your biggest competitive advantages when you're starting out, and you can figure out how to scale once you have your demand. And then maybe once you've scaled, then you go get that ice cream. Thank you

Q: How did your first customer hear about you?

A: Our very first one, I have no idea. We just launched in Palo Alto; we didn't do any marketing, so I assume he just must have typed in “Palo Alto delivery“ into the web browser. And then after that, we did barely any marketing. I think I sent out one email to my dorm, and that was about it. It was all through word-of-mouth. And that kind of just validates how strong the need we found was when people are just talking about you, and willing to put up with a terrible user experience, terrible design, and stuff like that.

Q: When you started, it seemed so obvious to you, you were wondering why, what the reason was nobody had done this before. What's your answer now looking back?

A: Looking back I think the biggest thing is mobile. Now everyone has one of those in their pocket, and we saw that trend and thought what if you could design a delivery system that was entirely based off mobile, where you didn't have to have any infrastructure, or delivery fleets. Instead of hiring drivers full-time or purchasing vehicles, what if you could tap into more of an on-demand pool of independent contractors, and only send orders to them when they have time. So that's kind of the insight we had; everything was done through mobile.

Q: Did you know you were going to be a startup, or were you just making some money at first?

A: At the time we were all just really passionate about building technology for small business owners, and obviously this delivery thing came out of an experiment with the landing page. It was literally an experiment. We weren't expecting anything, and it just took off, and we just went with it. And logistics was always something we were really passionate about as well, like logistics of transportation – the perfect fusion of how you can help small business owners through delivery.

Q: Did you launch the mobile site first or the website?

A: We started with this landing page right here which took us about an hour to launch.

Q: How does DoorDash stand out amongst a very competitive space?

A: At the beginning consumer demand was never a problem, even up until now. So for us it's just about finding a need and just focusing on serving that demand. At the beginning competition doesn't really matter.

Q: How long did it take you to get incorporated into a company?
A: We launched in January 2013, and then we did YC that very summer. When we decided to take this idea through YC, we incorporated.

Q: Where do you plan to go beyond food delivery?

A: For us when we started DoorDash, it was always about helping small business owners and figuring out how you served this for any local merchant whether you were a macaroon store, restaurant, or furniture shop. That's still our focus; that's our long-term vision. For now we are just focused on restaurant delivery as a way to scale, but ultimately that's where we want to end up in.
Walker Williams

Sam Altman: Next is Walker Williams, founder of Teespring. He's been working with YC for about a year and a half, something like that. I almost rejected him, which sounds like a dumb idea, but now they're making hundreds of millions of dollars in revenue, so very luckily I did not. Walker is also going to talk about doing things that don't scale.

Walker: Thank you guys for having me! My name is Walker; I am the CEO/Founder of Teespring. For those of you who don't know what Teespring is, we are an e-commerce platform that allows entrepreneurs to launch products and apparel brands without risk, cost, or compromise. Today the company is about 180 folks and we ship tens of thousands of products each day. I want to talk to you about one of the most fundamental advantages you have as a start up, and that's that you are able to do things that don't scale.

I define things that don't scale as things that are sort of fundamentally unsustainable; they will not last; they will not bring in the millionth user. Where they break, it's usually time but it could be a number of other things. But it's really growth strategies that won't take you to a million users. There are three places I want to focus on today. First one is finding your first users. The second one is turning those users into champions, and the third one is finding your product/market fit.

So finding your first users: The first thing you have to understand is that there's no silver bullet for user acquisition. You know, everybody, and this includes me when we got started, looks for that dream solution, that paperclip campaign that has tremendous ROI, some accelerating partnership that's going to springboard you into the stratosphere, and affiliate agreement; something that solves it for you. But the reality is for the vast majority of companies and in fact for every company that I've had the chance to speak to the CEO of, that's just not possible - those are unicorns. And most of the companies that from the outside look like they've had this dream growth curve, the reality is that those first users were impossibly hard to get. Let me tell you about the story about this ridiculously unsustainable business.

So this is Teespring in 2012 (PowerPoint). When we first launched, the business couldn't have looked worse. It took days of meetings; we had to offer free designs, and days of revisions back-and-forth, we'd have to launch the product ourselves, we'd have to do the social media, all to sell about 50 shirts to a local nonprofit and generate about $1000 of revenue. Anybody looking in would've said, "You guys have to give up, this is a terrible idea." But as time went on, those users started to add up, and I think something you have to understand is when you first launch a company, just by virtue of the fact that it's a new product, you're going to be bad at selling it right? You've got no idea what the pain points of customers really are. You've never sold that before. You don't have any success stories to point to, or testimonials. Those first users are always going to be the hardest.

And so it's your responsibility as a founder to do whatever it takes to bring in your first users. It's going to be different for every company. The common thread that I hear is, founders need to spend personal time and effort, a lot of their personal time and effort to bring those users in themselves. It could mean a number of things - anything from sending 100 emails a day, getting on the phone and just calling as many people as you can, going through a network like Stanford or Y Combinator. Anything you can do to just get that first user. I really equate it to pushing a boulder uphill. And if you think of a smooth hill when you get started, the incline is the steepest and those first inches are the hardest. But over
time as you get farther and farther, the incline steadies out, it gets easier, and eventually you reach a point where you're at the top of the hill and the boulder starts to roll on its own.

And so for those first users, you cannot just focus on ROI in the sense of time. Do not expect to spend an hour and return thousands of dollars. Maybe Stanley was one of those unicorns - really incredible story. But for most of us, those first two users are going to take a lot of handholding, a lot of personal love, and that's okay - that's essential for building a company. The one caveat of that is, I don't recommend giving your product for free. And there are plenty of exceptions to this rule, but in general, cutting costs or giving the product away is an unsustainable strategy I wouldn't recommend. You need to make sure that users value your product. And you know, people treat products that are free in a much different way than a paid product, and often times it can give you a false sense of security like, "Oh we're getting all these users; surely we can convert them to paid."

The second aspect is what happens when you get those users? How do you turn those users into champions? A champion is a user who talks about and advocates for your product. Every company with a great growth strategy has users who are champions. The easiest way to turn a user into a champion is to delight them with an experience they are going to remember, so something that's unusual or out of the ordinary – an exceptional experience.

The easiest way to do this early, and again something that is completely unsustainable - it's not going to scale forever - is to just talk to those users. People will say this all the time, it's sort of the core tenant of Y Combinator, is talk to users. I cannot stress how important it is that you spend a large chunk of your time talking to users. You should do it constantly, every single day, and as long as possible. Today at Teespring, I'm still the catchall email address, so anytime anybody misspells "support" or writes an email address that doesn't exist, I get that email. And so I still do about 10 to 20 customer service tickets every single day; I spend hours each night reading every single tweet, probably a little bit OCD, but that's okay; I read through all the Teespring communities. You're never going to get a better sense for your products than actually listening to real users. Especially in the early days, the product and the feature set you launch with is almost certainly not going to be the feature set that you scale with. So the quicker you talk to users and learn what they actually need, the faster you can get to that point.

There are three ways to talk to your customers. You can run customer service yourself. Up until Teespring was doing about $130-$140,000 a month, my cofounder Evan and I did everything in customer service. This is one where there's going to be an instinct to quickly pass off, and that's because it's painful. Even today when I open our customer service portal, I have an emotional reaction where my stomach sinks because it sucks talking to so many users who have had a terrible experience, and it's painful that something that you love and put so much effort into, to know you got it wrong or somebody didn't treat them right. But it's so important that you go through that and learn what you need to build, and what you need to fix.

The second step is to proactively reach out to current and churn customers. Churn customers are customers who have left. This is one that often falls by the wayside in the pursuit of new customers, but you want to make sure that your customers are having consistent good experiences; you don't want to take your current users for granted. When a user actually leaves your service, you want to reach out and find out why, both because that personal outreach can make the difference between leaving and staying; sometimes people just need to know that you care and it's going to get better. And even if you can't bring them back, there's a chance that you can learn from the mistakes you made that caused them leave, and fix it so you don't churn users out in the future in the same way.
Finally, the one I'm probably most OCD about is social media and communities. You need to know how people are talking about your brand. You need to try to make sure that when somebody does have a bad experience, and they're talking about it, that you make it right. Problems are inevitable: You're not going to have the perfect product; things are going to break; things are going to go wrong. That's not important. What's important is to always make it right, to always go the extra mile and make that customer happy. One detractor who's had a terrible experience in your platform is enough to reverse the progress of 10 champions. That's all it takes, is one to say, "No you shouldn't use those guys for X reasons," to ruin a ton of momentum.

There are examples in the early days where we would mess up massive orders. We'd print out colors slightly wrong; it would be the wrong size, and it would be half of our GMV for that month. We would know we got it wrong, and the customer would be unhappy, and the instinct was to say that it was only a little bit off, not completely wrong, or that it would be fine. But the reality is you just have to bite the bullet and make sure it's right. And the customers who are originally the most frustrated tend to turn into the biggest champions and the longest term users.

The last one I want to talk about is finding product/market fit. What I mean by that is the product you launch with will almost certainly not be the product that takes you to scale. So your job in those early days of a startup is to progress and iterate as fast as possible to reach that product that does have market fit. And as engineers your instinct is building a platform that's beautiful, clean-code, and that scales. You don't want to write a duct tape code that's going to pile on technical debt. But you need to optimize for speed over scalability and clean code. An example of this is in the early days, we had a couple enterprise customers come in, sort of bigger nonprofits, and say "Hey, we really like your service, but you're missing these fundamental things, so we're not going to use it." And we looked at what it would take to build out those features, and we weren't sure if they were going to work out long-term, but we wanted to try it.

My cofounder Evan, who is our CTO and a million times better developer than I am, ran the math and figured out that if we did it the right way, it was going to take about a month to build out these features. A month for a startup – you live in dog years – a month is a year, and that just wasn't going to do. So he actually went out and duplicated the code-base, duplicated the database, and was able to basically build a completely different product so that he didn't have to worry about the existing users to serve these enterprise customers. We gave them the tool, they on-boarded, and generated a lot of revenue. Eventually we learned what features were core, and we integrated them into the core product. But what would've taken a month, we were able to do in three to four days.

A great rule of thumb is to only worry about the next order of magnitude, so when you have your tenth user, you shouldn't be wondering how you are going to serve one million users. You should be worried about how you're going to get to 100. When you're at 100, you should think about 1,000. It's one of those things where necessity is the mother of invention, so when you hit the breaking point (the Twitter Fail Whale is a great example), and in Teespring there were month-stretches where every single night the site would crash - every night. Every single person on the team would go to sleep with their phone on loud, under their pillows, so inevitably when their phones went off, we could quickly restart the server and go back to sleep; this would happen daily. But the reality is that it was worth it, and you'll end up with these huge pain points and all this technical debt and regret, but it's worth it just to get to that end goal and that product fit faster. You will make it work; you will survive. Those bumps are just speed-bumps, and speed is so so important early.

The lesson that I've been learning lately is that you want to do things that don't scale as long as possible. There's not some magical moment; it's not Series A, or it's not when you hit a certain revenue milestone that you stop doing things that don't scale. This is one of your biggest advantages as a
company, and the moment you give it up, you're giving your competitors that are smaller and can still do these things, that advantage over you. So as long as humanly possible, as long as it is a net positive, you need to spend time talking to your users, you need to move as fast as possible in development, but don't give it up willingly; it should be ripped from you.

To practice what I preach, I want to give you guys my email address. If you guys have any questions, if you want to learn about Teespring, or if you want to print some T-shirts (fingers-crossed), just shoot me an email. I'd love to help and I'd love to speak to you.

Walker@teespring.com

The last thing is, we've created an official "How to Start a Startup" T-shirt with Sam. All proceeds are going to Watsi.org. I couldn't miss this opportunity to sell, so if you guys want to grab one of the official tees, just go to teespring.com/startup; it's supporting a great cause.

Thank you.

Q: What convinced you to get into the market of T-shirt printing when there is so much competition in this business?

A: I think there are two factors to it. First, I completely agree. From the outside, people have been telling us that this is a silly idea since day one in every order magnitude we reach. People will come and say, "This is a terrible idea. Why are you doing that?" But the reason why we launched Teespring is because we ran into a personal pain point where we had a need and looked at the current solutions. I was a student at Brown trying to create a "Remember the Bar" shirt for a dive bar that got shutdown, and I realized that nothing matched my needs. And so because I knew that I had that pain point, and I knew there was market fit, and I had seen people adopt the product, I knew there was something there. And it was also one of those things where you could sort of feel the wind on your back where people were adopting the product quickly. The pain point was clearly there; it's not a met need. So I would say that most times, great ideas start out by looking like the silly ideas, and then you can feel out whether or not there is a scalable business by how people are adopting it and whether it is possible to bring customers onboard.

Q: Are nonprofits your biggest customer base?

A: No, today our biggest customer base is entrepreneurs who are trying to build brands and businesses. We have a little over 1000 people who make their full-time living on Teespring today via brands they've launched. And the other side is influencers, so YouTube stars, Reddit communities, bloggers who want to add product merchandise as a way to create a brand and monetize that affinity. Those are our two biggest markets. We still work with a lot of nonprofits, and love working with them. They are still part of our business, but just not the majority.

Thank you.
Lecture 8: Doing Things That Don't Scale, PR, and How to Get Started

http://startupclass.samaltman.com/courses/lec08/

Justin Kan

Sam Altman: Now we have Justin Kan. Justin was the founder of Kiko, and the Justin.tv which became Twitch. He is going to talk about Public Relations.

Justin: I've started a lot of start-ups, but I think you've heard a lot of awesome "how I got started" stories, so I'm going to talk about something very specific that people always have questions about, which is press: how do you get it and how does it work? This is kind of like an abridged version of what we talk about at Y Combinator. Hopefully you guys will find it helpful.

When most people get started with entrepreneurship, they think about press and being in the press as something that happens magically. They think about journalists out there on the field trying to get the best stories - it's like a meritocracy - which is absolutely not the case.

Before you think about press, one of the things you really want to consider is who you want to reach, as well as your actual goal. I know when I got started I wanted to be in the news because I thought that's what you did as an important company. It turns out that if you don't have any goals, you're not going to achieve anything - that's true of pretty much everything. And with press, if you aimlessly just want to be covered, it's not going to do anything for your startup. If you don't have an actual business goal, then it's not a good use of time.

So there are many different goals. With Socialcam, which is a spinoff of Justin.tv, our goal was to be known as like video Instagram and be thought of in that context. When it was time to pitch to our Silicon Valley investors and influencers, we really wanted to get covered in tech press and be positioned as this hot, new social app.

With Exec (35:01) one of my goals was to get customers. Exec was like a local cleaning service, and our goal was to get people in San Francisco to use it. It wasn't useful to get national press because 99% of those people couldn't use it. So we targeted initially local press like SF Chronicle, who would directly talk to people who could potentially use our app. TwitchTV, which is probably what you guys mostly know, is like ESPN for a ___(35:33) kind of like a live stream community of gamers. Our goal was to reach the gaming industry. Now it's like 55 million ___(35:45) and people in the gaming industry know about it, but when we started nobody really knew that; we were a very small gaming community. Our goal was to get people in the gaming industry, whether they were developers or advertisers, to think about us as an important place where influencers were. So we really targeted industry trades and game ____ (36:06) blogs.

So what's an actual story? I think there's a bunch of different types of stories, but these are the ones you usually see in start-ups. Product launches are when you launch a different version of your app.

There's fundraising; the press loves to write about fundraising even though it's not very interesting. So like if you raise a million dollars __ (36:38) you can pretty much get that covered.

Milestones are metrics, like if you've achieved one million dollars a week in revenue. The company that bought Exec (36:49) just announced that they achieved one million dollars a week in revenues, and it was covered pretty widely. Business stories, which happen when you're a successful company, The New
York Times, The New Yorker, or Business Magazine, will want to cover your start-up story. You don't have to worry about that in the beginning.

What I like to call Stunts - I don't know if you guys remember, but a couple years ago, this YC company called WePay dropped a block of ice with money frozen outside of a PayPal developer's conference because PayPal was in the news for freezing various developers' account. It was widely covered because it was such an interesting thing, you know, it got them in the story in the context and all.

Hiring announcements: If you're a big enough company and you hire someone really important, people will want to cover that.

And finally contributed articles which would be you writing some sort of industry overview or some opinion piece, maybe a tech blog, stuff like that.

Basically any of those things can be a story. Something that people usually don't think about when you're trying to start a startup is you think that everything you're doing is interesting, but that may not be true for other people. What you really need to think about objectively is, if I wasn't the founder of this company, would I want to read a story about what I'm pitching? So your incremental feature release or your 2.01 feature release might not be interesting just because you added a "Find Your Contacts" app on Facebook. You really want to take a step back before you invest the time in actually trying to pitch a story, and think, "Will anyone actually really want to read this?" What journalists and bloggers are really looking for are things that people actually want to read.

The other thing is you don't actually have to be very original - your press doesn't have to be original. It just has to be what I like to call "original enough." You don't want to be the second-coolest company to raise $5 million on Kickstarter - the first guy gets all the news. The first video game console to raise $10 million on Kickstarter was huge news because they were the first in that category, even though a lot of people had raised a lot of money for Kickstarter before. Think about your stories in the context of other things that have been written and if they're like novel enough or aren't something that was just written about in the news.

So one of the actual mechanics of getting the story (this is pretty tactical), if you want to get your news in the press, basically there are some really easy steps to follow: Think of getting press like a sales funnel. You're going to talk to a lot of people but not all of them are going to convert – so you should be upset when one individual person or reporter doesn't write your story. The first thing is you have to think of it like a story. The second thing is you want to get introduced to any reporter or multiple reporters who are going to write about your thing. It's much much easier, just like any business development, to get in touch with them through someone, rather than cold emailing them.

The best thing to do is go to entrepreneurs who were just written about or friends who were just covered on TechCrunch, get them to introduce you to that reporter who wrote about them. The reason that's good is because from the entrepreneur's perspective, the easiest thing in the world to do is to introduce you to the reporter who already wrote about them. They don't need anything else from the reporter, and they're actually doing that person a favor if the story is interesting; it's not like you're asking for ____ (41:06) to investors or people they would want to hire like employees. And then from the reporter's perspective, they're getting introduced to someone who they already vetted as interesting; they're getting an answer from someone who they believed was interesting enough to write about, so by the transitive property, they're going to think you're probably interesting.

So you get an email from this guy who introduces you to the reporter, and you want to get in contact with them with enough time to get them to write a story - let them know probably a week in advance or...
more, because they're not going to drop everything they're doing to just write about your news. A lot of people, especially first time entrepreneurs, will come and say "Justin I'm launching this product tomorrow. Can you get me in TechCrunch?" That's probably not going to happen unless you already have a relationship. The best thing to do is give yourself some lead time; get that intro in advanced.

So once you've set a date for your news to go out – you're going to launch a product out in two weeks - you have this intro, and you've set up some sort of meeting, you really want to get the reporter to invest time and effort into you because there's some cause fallacy at play. Basically, the more time they spend with you, the more likely they are to write about you. The best thing to do is get a face-to-face meeting. Some bloggers don't actually want to meet face-to-face, but like if not that, then get a phone call. The worst thing to do is just have an email exchange because it's very easy for them to forget about it or ignore it.

The next step is actually pitch them. What I do is actually write out the ideal story that I want to see published in bullet points, and memorize it. And when I have a conversation with them if it's in person, the conversation is structured like my outline, and they'll be taking notes. Then they'll go transcribe those notes into a story. So basically what I wrote will be translated into an actual story. By preparing, you can actually control more of the conversation and not forget critical things like mentioning your cofounder's name or the awesome features in your app are. If I'm doing this over the phone, I'll make sure to have the bullet point list in front of me and I will make sure to walk through a conversation that includes all those things. So you have a pitch, they take notes, and they're going to write the story at this time. The next thing to do is follow-up like a couple days or day before your actual news goes out. You want to send them an email that says "This is time I'm launching the app; thanks for meeting; here's collateral if there are any videos, photos, or screenshot you want to include; how to spell your cofounder's name or your name. I just include all the information that I really care about and bold it. And then that's it. Hopefully the day comes when you press submit on the release to the App Store and at the same time, they released their article on TechCrunch and you are famous.

So a lot of people ask us about PR firms. So I think in the beginning it's kind of like, everything you do in a startup, you want to do yourself. And it's actually pretty easy especially with tech press and bloggers who constantly need new things to write about. I strongly encourage people to try it themselves, and kind of get started by learning the process themselves before they hire anyone. One thing I'll say is that firms can only help you with the contacts and the logistics, but they can't help you know what's interesting about your company, or – I'm never been able to have any one that's told me what the stories that I'm producing on; they've only been able to give me a list of reporters I might want to contact. So you really have to be responsible about thinking what's interesting about your company and what you're doing, you know, the roadmap of interesting things that you're working on.

They're also really expensive. I think we were spending about $5,000 and $20,000 a month, and for various firms, it's a lot - for a startup right? It's generally not a good use of money especially in the early days. Getting presses is a lot of work, so you should really make sure it's worth it. Getting presses is like a vanity metric: It feels like you're being successful because many successful companies like Facebook are covered in the press all the time, but it doesn't actually mean you're successful; it doesn't actually mean you're making money, getting users, or making those users happy. Sometimes it's a really good strategy for getting your first hundred or two-hundred or a thousand customers, but it's really not a user-scalable acquisition strategy, so it's really something that's just a bootstrap; you can't just get like infinity articles written about you. Eventually people are going to get tired of hearing about your company, and usually that happens really quickly right? The pull point about news is that it's new, so it's pretty hard, unless you're like Google to get covered in the press every week. If you decide it's worth it though, that you do want to have a regular heartbeat of news, you're thinking about what you're doing that matches those seven story types in the future. When I was working primarily on marketing
and PR, I would make a schedule on the calendar of when we're going to launch things and make sure to space them out, but have them appear at regular intervals so people didn't want to forget about us and we could maximize our coverage.

You also want to keep your contacts fresh; it's really a relationship business. Once someone writes about you, you should keep up with them because they could potentially write about you in the future. You're more likely to do something for someone that you've already done something for in the past. I would try to establish good relationships with a couple reporters that you could go to for breaking news; it could come in handy later if you're fortunate enough to have people writing negative things about you; your relationships will help you get your side of the story.

The last thing is kind of like a golden rule or more of like a "pay it forward:" You should help your fellow entrepreneurs get coverage because they'll help you get coverage. The best way to get coverage is really through these introductions. Whenever I'm meeting with reporters, I throw out names of other things I think would be interesting stories for them, and usually that comes back. The reporters like it because you're helping them find interesting stories, and you're more likely to get leads back from those entrepreneurs that you helped out.

If you're interested in learning more about press, here are two resources that I really like: Jason Kincaid, who is a former TechCrunch reporter, goes through a great overview that covers a lot of things I talked about in more depth. And then kind of an evil resource is this book called Trust Me, I'm Lying which was written by a former marketer at American Apparel. He talked about ways that he evilly manipulated the press, but I think it's a pretty good look into the psychology of how stories spread on the Internet; it's valuable to take a look at. Cool, that's basically it.

Q: When is the right time to start worrying about press altogether

A: The first time I launched my first start up, you know my first products, for a lot of them we got zero attention and we didn't really know how we could get 100 users. I think it's a really fine way to get 100 users, and a lot of companies at YC when they first launch their product, we'll encourage them to get out and do one TechCrunch story to get a few people to see it. It's good to get in the practice. I wouldn't obsess getting coverage in multiple outlets or anything like that in the very beginning.

Q: (50:37 low quality audio) How much of a role did you guys play in getting the Pokemon thing out?

A: Twitch had this thing called "Twitch Plays Pokemon" where developer set up a Pokémon Gameboy game that was controlled by chat, so millions of people would be typing in A or B, and the character would wander around aimlessly. That was a huge news story. I think what we did was set the stage by having other news stories that someone from the BBC would Google "Twitch" and be like, "What is this crazy thing that everyone on Reddit is talking about?" They would have some context. The other thing is we didn't come up with the idea for Twitch. That was like fortuitous, but we helped give it legs by making the company available to talk to reporters and suggesting follow-up stories, not only about "Twitch Plays Pokémon" with 100,000 people watching, but also stories about when they beat the game or when Twitch played Crystal, the new Pokémon version. We didn't originate the story; it was the community who I think really originated it.

Okay I think that's it, thank you very much!
Lecture 8: The Press is a Tool

https://docs.google.com/document/d/1LQxnHxQ6xO54BHcoOmgEeuhdHwWwujKuSpzQbQnIThk/edit?pli=1

Alexia Tsosis

A TechCrunch article written specifically for Sam Altman’s "How To Start A Startup" CS138B Course

From the outset, you should know why you’re courting media coverage. Press can be important for getting early customers/users, receiving resumes for potential hires, creating awareness among investors, boosting the morale of your team, raising awareness or providing social proof for potential business development partners.

Especially in the case of e-commerce/transactional consumer companies, garnering press (I believe the term is “earned media”) can be one of the sharpest ways to acquire customers and generate revenue early on. As an early-stage startup, free is better than not free, especially when you can fall into the trap of spending too much to acquire the wrong kind of customers.

At its most basic level, press is about telling stories. As a nascent company, think of your media outreach as practice for when you run a multi-billion dollar business and you have to tell stories to not only the media, but also your 30,000 employees, shareholders, and potential shareholders.

I know I’m very biased, but building relationships with press people early on can pay dividends, depending on the nature of your company and your strategy. Solid press contacts will follow your company for its lifetime.

Here are four “dealing with the press” strategies and outcomes I’ve seen work and not work for founders:

1) A startup gets no press, gets no traction, dies.
2) A startup gets press, gets no traction, dies.
3) A startup gets press, gets traction, goes public, or gets bought by Facebook (or Google, or Yahoo, etc).
4) A startup gets no press, gets traction, goes public or gets bought by Facebook (or Google, or Yahoo, etc).

As my former TechCrunch colleague Jason Kincaid points out in his book, “A Burned Out Blogger’s Guide To Startup PR” what situation founders experience is largely random. “This is all a big lottery,” he says, “and press is a good way to get more tickets.” (You can probably stop reading this and just go buy Jason’s book. From explaining the best way to deal with embargoes — when multiple reporters wait to publish your story — and defining vague terms like “off the record” versus “on background” the ideas in the book might save you a lot of money on formal PR, which at its best and most expensive is really good, but at its worst is basically like burning money.)

You may have an amazing product that people will discover eventually, or a terrible product that everyone knows about. Sometimes a John Mayer tweet really is the difference between ending up as “Words With Friends” — acquired by Zynga for more than $53 million — and ending up a casualty of the App Store, consumer tech’s boulevard of broken dreams.

Of all the outcomes on that list above, probably the worst one is #2, a startup that’s over hyped in the press but still hasn’t quite figured out how to make something that people

A TechCrunch article written specifically for Sam Altman’s “How To Start A Startup” CS138B Course

From a startup’s perspective the press is a tool (And it’s a tool that doesn’t want to get used!). Dealing with press is part of the job of building a successful tech company. But just like you don’t want to live in a house made of hammers, press mentions shouldn’t by any means be the only piece of your growth strategy -- or be interpreted as “Great, we’re successful now, so we can slow down.”

If it’s needed at all to reach your goals as a startup, press is a necessary evil (like raising capital). It can be a massive accelerator, an awesome catalyst towards exponential opportunities when approached with the right mindset. It can also be an unnecessary distraction. Theranos received its first press after it had already become a $9 billion behemoth -- sprung fully formed from the head of Elizabeth Holmes. And WhatsApp became Facebook’s largest buy with little more than a peep from the press.

In this class you may have already been discouraged from approaching the media until you get traction. Do us both a favor and take that advice. :)

But, if you can’t resist orchestrating a splashy company launch, or somehow know a reporter, or for whatever reason want to dabble in the dark arts of press; here are some of my observations in over six years of covering our industry -- three as co-editor of TechCrunch, an almost decade old publication that is still, miraculously, relevant.

In this class you may have already been discouraged from approaching the media until you get traction. Do us both a favor and take that advice. :)

But, if you can’t resist orchestrating a splashy company launch, or somehow know a reporter, or for whatever reason want to dabble in the dark arts of press; here are some of my observations in over six years of covering our industry -- three as co-editor of TechCrunch, an almost decade old publication that is still, miraculously, relevant.

From the outset, you should know why you’re courting media coverage. Press can be important for getting early customers/users, receiving resumes for potential hires, creating awareness among investors, boosting the morale of your team, raising awareness or providing social proof for potential business development partners.

Especially in the case of e-commerce/transactional consumer companies, garnering press (I believe the term is “earned media”) can be one of the sharpest ways to acquire customers and generate revenue early on. As an early-stage startup, free is better than not free, especially when you can fall into the trap of spending too much to acquire the wrong kind of customers.

At its most basic level, press is about telling stories. As a nascent company, think of your media outreach as practice for when you run a multi-billion dollar business and you have to tell stories to not only the media, but also your 30,000 employees, shareholders, and potential shareholders.

I know I’m very biased, but building relationships with press people early on can pay dividends, depending on the nature of your company and your strategy. Solid press contacts will follow your company for its lifetime.

Here are four “dealing with the press” strategies and outcomes I’ve seen work and not work for founders:

1) A startup gets no press, gets no traction, dies.
2) A startup gets press, gets no traction, dies.
3) A startup gets press, gets traction, goes public, or gets bought by Facebook (or Google, or Yahoo, etc).
4) A startup gets no press, gets traction, goes public or gets bought by Facebook (or Google, or Yahoo, etc).

As my former TechCrunch colleague Jason Kincaid points out in his book, “A Burned Out Blogger’s Guide To Startup PR” what situation founders experience is largely random. “This is all a big lottery,” he says, “and press is a good way to get more tickets.” (You can probably stop reading this and just go buy Jason’s book. From explaining the best way to deal with embargoes — when multiple reporters wait to publish your story — and defining vague terms like “off the record” versus “on background” the ideas in the book might save you a lot of money on formal PR, which at its best and most expensive is really good, but at its worst is basically like burning money.)

You may have an amazing product that people will discover eventually, or a terrible product that everyone knows about. Sometimes a John Mayer tweet really is the difference between ending up as “Words With Friends” — acquired by Zynga for more than $53 million — and ending up a casualty of the App Store, consumer tech’s boulevard of broken dreams.

Of all the outcomes on that list above, probably the worst one is #2, a startup that’s over hyped in the press but still hasn’t quite figured out how to make something that people
want. Every time you court press, you have more than a 25% chance (much more when weighted) of getting this outcome. Scary.

Not many startups can say “Look what we did.” It’s impossible to get metrics for unlaunched products and as a startup you’re judged from where you’re at today — nowhere.

On the bright side, as a startup, people give you the benefit of the doubt, but once once you pass about the $5 million valuation mark (essentially the moment you graduate from YC) people expect you to put your traction where your mouth is.

Just like how the outsized Price/Equity ratios for public tech companies like Facebook and Twitter can be good because they signal room for future revenue growth, hype in private markets can be good because it = Potential.

Some companies that have used the hype cycle successfully are Uber, Twitter and Airbnb. You might have heard of them. But when a company reaches a peak, hype-wise, it’s time to start bearing fruit. Fruits can be anything from exits like Instagram to going public like Twitter to a high valuation like Uber’s exceptional $12 billion. Or the old mainstay: revenue results.

While everyone hopes for a Cinderella story, truth is that over 90% of startups fail. 74% percent of these startups fail because of premature scaling (i.e. the hype overwhelming product capabilities). Companies that scale properly also attract more growth, capital and customers, and eventually hire more employees. Companies that don’t ...

End up like Color. Color raised $41 million from Sequoia Capital, with the WSJ saying “Not since Google have we seen this ...” in its funding announcement.

Color had a lot of hype, but within two years had sold to Apple for less than 1/8 of its splashy funding round. Sequoia has since notched the biggest acquisition exit in venture history with Whatsapp, but I will never forget Color. Imagine how much good that money could have done invested elsewhere, like solving the SF housing crisis perhaps, or in your startup?

Despite the cautionary tales, hype, as random as it is, is still the currency of Silicon Valley. People think Silicon Valley is a magical place filled with amazing technologies. But we also have some of the best marketers in the world: RIP Steve Jobs.

When you’ve got nothing, hype, manufactured by blog posts and marketing gimmicks actually can help raise VC and capture the imagination of early adopters. And sometimes it really works out for startups — Outcome #3: The over-hyped Mailbox was indeed picked up for $100 million by Dropbox, though it was flawed (it didn't send attachments for example). It might live up to the hype, though, and that's where the magic happens.

Path, Foursquare and even Square each have a brand in the market that is further along than the demand for their products, which creates expectations from press, investors and users. And hype can sometimes actually push away the help that the startups need. A large over-hyped round raises expectations, a small one lowers them. VC Vinod Khosla has asserted, "Hype may help companies get investments faster, but at the same time may scare away investors (at too high of a valuation)."

Crunchbase backs this up, we took data from 20 overhyped startups (Path, Quora) whose products haven’t gained exceptional traction and compared them to successful, hyped up startups like Instagram and Twitter.

Crunchbase data holds that hyped-up failures raise quicker than hyped-up successes in Seed or Series A funding, but then have a harder time raising Series B. While hyped/successful companies do end up raising more money in Series B, this may be attributed to a well defined product after the initial funding, which ultimately gets them more funding thereafter.

For the past five years, the mythological desired outcome (#3) is such: Get your TechCrunch post, get 1 million users, raise a huge round, sell to Google for $100 million. Now that narrative is changing — Get your Product Hunt link, TechCrunch post, Re/Code post, WSJ post, blog on Techmeme and HackerNews, sell. But the average experience, even when you do sparkle, is fade.

So yeah, you don’t need the press, or at least as much as you think you do.

Apart from its initial seeding to bro bloggers, for the longest time, Snapchat tried to be stealth on purpose. And there are a ton of successful companies that have never been covered by TechCrunch. Or, like Whatsapp, were not covered extensively. Especially those launched before 2005. We didn’t exist back then.

And if you’re an enterprise company, it especially doesn’t matter.

So what’s the best media/hype strategy through all the risk and randomness? Go with Warren Buffet: “Markets are risky, good businesses are not.”
Whatever your position is in the press cycle, maintain a positive attitude and design your business intelligently, while constantly asking yourself:

1) What problem am I solving for my customers?
2) Does my startup have a reason to exist?
3) How can I make my service even better?
4) Am I improving things for the economy and society at large?

And as an aside, the three questions I always ask founders in interviews are:

1) Who is your closest competitor and what do you do differently?
2) What are the challenges of doing this?
3) What are your future plans?

One simple rule in hype as in life is avoid over-promising and under-delivering.
If you ever need anything from me, or just have more questions: My email is alexia@techcrunch.com
9. How to raise money
Marc Andreessen

Sam Altman: I would like to start with a question for Mark and Ron, which is by far the number one question we are going to be looking at today, what made you decide to invest in a founder company?

Ron Conway: Well we have a slide on that-

Sam Altman: Mark can start while we get that up.

Ron Conway: So what makes us invest in a company is based on a whole bunch of characteristics. I have been doing this since 1994, right before Mark got out of the University of Illinois, so SV Angel and its entities have invested in over 700 companies. To invest in 700 companies that means we have physically talked to thousands of entrepreneurs and there is a whole bunch of things that just go through my head when I meet an entrepreneur. I am just going to talk about what some of those are.

Literally while you are talking to me in the first minute I am saying “Is this person a leader?” “Is this person rightful, focused, and obsessed by the product?” I am hoping-because usually the first question I ask is "What inspired you to create this product?"-I’m hoping that it’s based on a personal problem that that founder had and this product is the solution to that personal problem. Then I am looking for communication skills, because if you are going to be a leader and hire a team, assuming your product is successful, you have to be a really good communicator and you have to be a born leader. Now some of that you may have to learn those traits of leadership but you better take charge and be able to be a leader. I’ll switch back to the slide, but let's let Mark.

Marc Andreessen: Yeah, I agree with all of that. There is a lot of detail to this question that we can talk about. And we may be a little different than Ron, well we are different than Ron, in that we invest in across stages. We invest in the seed stage, the venture stage, growth stage. And then we invest in a variety of business models: consumer, enterprise, and a bunch of other variations. There are a bunch of fine grained answers that we could get into if there are specific questions.

Two general concepts I would share: one is the venture capital business is one hundred percent a game of outliers, it is extreme outliers. So the conventional statistics are in the order of four thousand venture fundable companies a year that want to raise venture capital. About two hundred of those will get funded by what is considered a top tier VC. About fifteen of those will, someday, get to a hundred million dollars in revenue. And those fifteen, for that year, will generate something on the order of 97% of the returns for the entire category of venture capital in that year. So venture capital is such an extreme feast or famine business. You are either in one of the fifteen or you’re not. Or you are in one of the two hundred, or you are not. And so the big thing that we’re looking for, no matter which sort of particular criteria we talked about, they all have the characteristics that you are looking for the extreme outlier.

The other thing I would highlight that we think about a lot internally, we have this concept, invest in strength versus lack of weakness. And at first that is obvious, but it’s actually fairly subtle. Which is sort of the default way to do venture capital is to check boxes. So really good founder, really good idea, really good products, really good initial customers. Check, check, check, check. Okay this is reasonable, I’ll put money in it. What you find with those sort of checkbox deals, and they get done all the time,
but what you find is that they often don’t have something that really makes them really remarkable and special. They don’t have an extreme strength that makes them an outlier.

On the other side of that, the companies that have the really extreme strengths often have serious flaws. So one of the cautionary lessons of venture capital is, if you don’t invest in the bases with serious flaws, you don’t invest in most of the big winners. And we can go through example after example after example of that. But that would have ruled out almost all the big winners over time. So what we aspire to do is to invest in the startups that have a really extreme strength. Along an important dimension, that we would be willing to tolerate certain weaknesses.

Ron Conway: Okay, I don’t want to over dwell on the slide, but when you first meet an investor, you’ve got to be able to say in one compelling sentence that you should practice like crazy, what your product does so that the investor that you are talking to can immediately picture the product in their own mind. Probably twenty-five percent of the entrepreneurs I talk to today, still after the first sentence, I don’t understand what they do, and as I get older and less patient I say “Backup, I don’t even know what you do yet.” So try and get that perfect. And then I want to skip to the second column. You have to be decisive, the only way to make progress is to make decisions. Procrastination is the devil in startups. So no matter what you do you got to keep that ship moving. If it’s decisions to hire, decisions to fire, you got to make those quickly. All about building a great team. Once you have a great product then it’s all about execution and building a great team.

Sam Altman: Parker, could you talk about your seed round and how that went and what you should have done differently for raising money?

Parker Conrad: Sure, actually my seed round, most of the stuff with my current company felt like, from a fundraising perspective, felt like it came together relatively quickly. But actually, one of the experiences I had, I started a company before this that I was at for six years, and my co-founder and I pitched almost every VC firm in Silicon Valley. We literally went to sixty different firms and they all told us no. And we were constantly trying to figure out how should we adjust our pitch? How should we do the slides differently? How do we tweak the story? That sort of thing. At one point there was this key insight that someone gave me when I was pitching at Khosla Ventures and this VC said “Guys,” he was looking for some very particular kind of analysis that we did not have on hand, he was like “Guys, you don’t get it.” He was like, “You know if you guys were the Twitter guys, you guys could come in and just be blublublubluh and put whatever up here and we would invest in you. But you guys aren’t the Twitter guys so you need to make this real easy and have all this stuff ready for us,” and all this kind of stuff.

And I took the exactly opposite lesson of what he wanted me to take away from that which was: geez I should really just figure out how to be the Twitter guys and that’s the way to do this. So actually one of the reasons I started my current company or one of the things that I found really attractive about Zenefits is, as I was thinking about it, it seemed like a business. I was so frustrated from this experience of having tried for two years to raise money from VCs and I sort of decided, to hell with it. You cannot count on there being capital available to you. This business that I started seemed like one that maybe I could do without raising money at all. There might be a path to kind of, there’s enough cash flow it seemed compelling enough that I could do that.

It turns out that those are exactly the kinds of businesses that investors love to invest in and it made it incredibly easy. So I actually think it seems very kind and I said I was an expert for you guys. I don't actually think I am very good at fundraising. It is probably something that I am less good at than other parts of my job. But I think that if you can build the business where everything is moving in the right direction, if you could be like the Twitter guys, like nothing else matters, and if you can’t be like the
Twitter guys, then it is very hard for anything else to make a difference for things to come together for you.

Ron Conway: Why did that VC say to be like the Twitter guys when the fail whale dominated that site for two years?

Marc Andreessen: Because it worked.

Ron Conway: The other point I want to make is, bootstrap for as long as you can. I met with one of the best founders in tech who's starting a new company and I said to her “Well, when are you going to raise money?” "I might not," and I go, "That is awesome." Never forget the bootstrap.

Marc Andreessen: So I was actually going to close on this, but I'm just going to accelerate it. Parker, I think, gave you the most important thing you will ever hear. Which is also what I was going to say. So the number one piece of advice that I have ever read and that I tell people on these kind of topics is always from the comedian Steve Martin, who I think is an absolute genius, wrote a great book on the start of his career, which obviously was very successful. The book is called “Born standing up”, it's a short little book and it describes how he became Steve Martin. And the part of the book is, he says what is the key to success? The key to success is be so good they can't ignore you.

So in a sense, we are going to have this whole conversation and I am sure we will keep having it, but it is beside the point, because if you do as Parker has done and you build a business that is going to be a gigantic success then investors are throwing money at you. And if you come in with a theory and a plan and no data and you are just one of the next thousand, it's going to be far, far harder to raise money. So that is the positive way to put it, is to be so good they can't ignore you. You are almost always better off making your business better than you are making your pitch better.

The other thing, that's the positive way to look at it, the negative way or the cautionary lesson is that, and this gets me in trouble every single time I say it, but I am on a ton of flu medications so I am going to go ahead and just let it rip, raising venture capital is the easiest thing a startup founder is ever going to do. As compared to recruiting engineers, recruiting engineer number twenty. It's far harder than raising venture capital. Selling to large enterprise is harder, getting viral growth going on a consumer business is harder, getting advertising revenue is harder. Almost everything you'll ever do is harder than raising venture capital. So I think Parker is exactly right, if you get in the situation where raising the money is hard, it's probably not hard compared to all the other stuff that is about to follow. It is very important to bear that in mind. It's often said that raising money is not actually a success, it's not actually a milestone for a company and I think that is true. And I think that is the underlying reason, it puts you in a position to do all the other harder things.

Sam Altman: Related to that. What do you guys wish founders did differently when raising money? And specifically, Mark, you mentioned this relationship between money and funding?

Marc Andreessen: I think the single biggest thing that people are just missing and I think it's all of our faults, we are all not talking about it enough, but I think the single biggest thing entrepreneurs are missing both on fundraising and how they run their companies is the relationship between risk and cash. So the relationship between risk and raising cash, and then the relationship between risk and spending cash. So I have always been a fan of something that Andy Rachleff taught me years ago, which he calls the onion theory of risk. Which basically is, you can think about a startup like on day one, as having every conceivable kind of risk and you can basically make a list of the risks. So you've got founding team risks, are the founders going to be able to work together; then you have product risk, can you build the product; you will have technical risk, maybe you need a machine learning
breakthrough or something. Are you going to have something to make it work, or are you going to be able to do that? You will have launch risk, will the launch go well; you will have market acceptance risk, you will have revenue risk. A big risk you get into with a lot of businesses that have a sales force, is that can you actually sell the product for enough money to actually pay for the cost of sales? So you have cost of sales risk. If you are a consumer product, you have viral growth risk. So a startup at the very beginning is just this long list of risks, right, and the way I always think about running a startup is also how I think about raising money. Which is a process of peeling away layers of risk as you go.

So you raise seed money in order to peel away the first two or three risks, the founding team risk, the product risk, maybe the initial watch risk. You raise the A round to peel away the next level of product risk, maybe you peel away some of the recruiting risk because you get your full engineering team built. Maybe you peel away some of your customer risk because you get your first five customers. So basically the way to think about it is, you are peeling away risk as you go, you are peeling away your risk by achieving milestones. And as you achieve milestones, you are both making progress in your business and you are justifying raising more capital. Right?

So you come in and pitch to someone like us. And you say you are raising a B round. And the best way to do that with us is to say I raised a seed round, I achieved these milestones. I eliminated these risks. I raised the A round. I achieved these milestones. I eliminated these risks. Now I am raising a B round. Here are my milestones, here are my risks, and by the time I raise go to raise a C round here is the state I will be in. And then you calibrate the amount of money you raise and spend to the risks that you are pulling out of the business. And I go through all this, in a sense that sounds obvious, but I go through this because it is a systematic way to think about how the money gets raised and deployed. As compared so much of what's happening these days which is "Oh my god, let me raise as much money as I can, let me go build the fancy offices, let me go hire as many people as I can." And just kind of hope for the best.

Ron Conway: I'm going to be tactical. For sure don't ask people to sign an NDA. We rarely get asked any more because most founders have figured out that if you ask someone for a NDA at the front end of the relationship you are basically saying, I don't trust you. So the relationship between investors and founders involves lots of trust. The biggest mistake I see by far is not getting things in writing. You know, my advice on the fundraising process is do it as quickly and efficiently as you possibly can. Don't obsess over it. For some reason, founders get their ego involved in fundraising where it is a personal victory. It is the tiniest step on the way as Mark said. And it's the most fundamental. Hurry up and get it over with.

But in the process, when somebody makes the commitment to you, you get in your car, and you type an email to them that confirms what they just said to you. Because a lot of investors have very short memories and they forget that they were going to finance you, that they were going to finance or they forget what the valuation was, that they were going to find a co-investor. You can get rid of all that controversy just by putting it in writing and when they try and get out of it you just resend the email and say excuse me. And hopefully they have replied to that email anyways so get it in writing. In meetings take notes and follow up on what's important.

Sam Altman: I want to talk a little more about tactics here. Just how does the process go? Can people email you directly or do they need to get an introduction? And how many meetings does it take to make a decision? How do you figure out what the right terms are? When can a founder ask you for a check?

Marc Andreessen: That was six questions. It was a lot of things. Why don't you describe, you will describe seed then I will describe-
Ron Conway: Yeah so, SV Angel invests in seed stage startups, so we like to be the very first investor. We normally invest today at around the million to two million. It used to only be a million. So if we invest two hundred and fifty k, that means there’s five or six other investors in that syndicate. SV Angel has now a staff of thirteen people. I do no due diligence anymore, I am not a picker anymore. I just help on major projects for the portfolio companies that are starting to mature. But we have a whole team that processes. We, at SV Angel, end up investing in one company for every thirty that we look at, and we end up investing at about one a week.

I think what’s interesting is, we don’t really take anything over the transom. Our network is so huge now that we basically just take leads from our own network. We evaluate the opportunity, which means you have to send in a really great short executive summary and if we like that, we actually vote, although I am not in this meeting anymore, but the group actually takes a vote on do we make this phone call. That’s how important time is in this process. And if enough of the team at SV thinks it’s interesting then they appoint a person to make a phone call to that founder. Usually somebody on our team that has domain experience. If the phone call goes well, bingo! We want to meet you. If SV Angel asks you for a meeting, we are well on our way to investing. If that meeting goes well, we’ll do some background checks, back door background checks, get a good feeling about the company, the market that they are going after and then make the commitment to invest. And then start helping get other value-add investors to be part of the syndicate. Because if we are going to have an equal workload we want the other investors in this company to be great angel investors as well.

Marc Andreessen: Okay, so I will talk a little about the venture stage, the Series A stage. I think it’s fair to say at this point, the top tier venture capitalists only invest in two kinds of companies at the Series A stage. One is if they have previously raised a seed round. So it’s almost always the case when we are doing a Series A investment for the company when the company has a million or two million in seed financing, from Ron and the folks that he likes to work with. Almost always Ron, just to be clear and folks he likes to work with. So if you are going Series A, the first thing you to do is to raise seed, that is generally the way the progression works at this point. Every once in a while we will go straight to a company that hasn’t raised a seed round. Really the only times when that happens is when it is a founder who has been a successful founder in the past and is almost certainly somebody we have worked with in the past.

So actually, we have not announced, but we just did one of these we will announce in a few weeks where it was a founder, I was an Angel investor, actually I think Ron was also in the company in 2006. Then the company did it’s thing and ultimately was acquired by another big company. And now that team is now starting new things. So in that case we are just going to jump it straight to an A. Because they are so well known and they have a plan all lined up for it. That’s the exception, it’s almost always preceded by a seed round. The other thing is, I mentioned this already, but we get similar to what Ron said, about two thousand referrals a year through our referral network. A very large percentage of those are referrals through the seed investors. So by far the best way to get the introductions to the A stage venture firms is to work through the seed investors. Or work through something like Y Combinator.

Sam Altman: Speaking of terms. What term should founders care most about? And how should founders negotiate?

Parker Conrad: Probably precisely because of what Mark said, the most important thing at the seed stage is picking the right seed investors because they are going to lay the foundation for future fundraising events. They’re going to make the right introductions, and I think there is an enormous difference in the quality of an introduction. So if you can get a really good introduction from an someone that the venture capitalist really trusts and respects, the likelihood that that is going to go well is so much higher than a lukewarm introduction from someone they don’t know as well. So the seed stage, probably
the best thing you can do is find the right investors and then-

Sam Altman: How does the founder know who the right investors are?

Parker Conrad: Well, I think it’s really hard. I think one of the best ways, and not to give a plug to YC, is YC does a very good job at telling you who they think those people are. And can really direct you towards, and I actually found it to be pretty accurate in terms of who you guys have said were going to be the best people, they ended up being the most helpful as we were raising subsequent rounds that provided the best introductions. The people who I thought seemed okay, but were not as highly rated by YC that ended up being the case that they were real duds in the seed round.

Sam Altman: Someday we are going to publish some of these people-

Parker Conrad: Oh gosh there are going to be a lot of upset people if you do.

Sam Altman: So how do you think about negotiation? How do you figure out what the right evaluation of their company should be, what are the terms?

Parker Conrad: Well when I was starting out, I was raising my seed round and I didn’t really know. I mean, we had conversations about this. I probably started a little too high on the valuation side. As you guys know, Y Combinator starts this thing called Demo Day. You get all these sort of investors at once who are looking at the company. I started out trying to raise money for a twelve or fifteen million dollar cap. Which is not quite the same thing as a valuation but roughly the equivalent. And everyone thought that was crazy, you know, that's completely nuts. You are too big for your britches, that completely just wouldn't work. So I started working it down a little bit. Within the space of a couple days I decided I was going to raise at nine, and for some reason that hit some magical threshold on the seed stage that it was below ten that it seemed that there was almost infinite demand for the round at a nine million cap. So no one would pay twelve but at a nine million dollar cap it felt like I could have raised ten million dollars. And the round came together in roughly about a week at that point, once I hit that threshold.

There seemed to be, and they fluctuate over time, these thresholds particularly for seed stage companies that investors think that above this level is crazy, it doesn't matter. And there is a rough kind of range that people are willing to pay. You just have to figure out what that is. Just get the money that you need, don't raise any more than you need. And just get it done. At the end of the day, whether you raise a twelve, a nine, or a six, it's not a huge deal for the rest of the company.

Sam Altman: Is there a maximum in the company that you think founders should sell in their seed round, their A round? Beyond what Paul was talking about.

Parker Conrad: I don't know the rules on this stuff. The tricky thing is, it seems like they are particularly rough for a Series A. You are probably going to sell somewhere between twenty to thirty percent of the company. Below, venture capitalist tend to be a lot more ownership focused than price focused. So you might find that it's actually, when companies raise really big rounds it is because the investors says, "Hey listen, I am not going to go below twenty percent ownership but I will pay more for it." Above thirty percent, probably weird things happen with the cap table, like it gets hard, you know, down the line for there to be a firm on the cap table for everyone. Everything seems to come in in that range, so that probably just is what it is. In most cases, in the seed stage from what I have heard, there doesn’t seem to be any magic to it, but it seems ten to fifteen percent is what people say, but that is mostly just what I have heard.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Ron Conway: I agree with all of that. I think it is important to get the process over with. But I think it is important for the founder to say to themselves in the beginning, at what point does my ownership start to demotivate me? Because if there is a forty percent dilution in an Angel round, I have actually said to the founder, do you realize you have already doomed yourself? You are going to own less than five percent of this company if you are a normal company. And so these guidelines are important. The ten to fifteen percent is because if you keep giving away more than that there is not enough left for you and the team. You are the ones doing all the work.

Marc Andreessen: We'll actually, we'll walk. We have seen a series of interesting companies in the last five years that, where we just walk, we won't bid simply on the basis that their cap table is already destroyed. Outside investors already own too much. There is a company we really wanted to invest in, but the outside investors already owned eighty percent of it when we talked to them. And it was a relatively young company, they just had done two early rounds that sold too much of the company. Literally, we were worried, and accurately so, it was going to be demotivating for the team to have that structure.

Sam Altman: One more question before we open up for the audience to ask questions. For Ron and for Mark. Could you both tell a story about the most successful investment you ever made and how that came to happen?

Parker Conrad: Other than Zenefits.

Ron Conway: For me clearly, it was the investment in Google in 1999. And we got a googol return out of it. Funny enough, I meet Google through a Stanford professor David Cheriton, who is in the school of engineering and he is still here. He was actually an Angel investor in Google and an investor in our fund. Kind of the quid pro quo we have with our investors in the fund is you have to tell us about any interesting companies that you see. We loved it that David Cheriton was an investor in our fund because he had access to the computer science departments deal flow. And we were at this party at Vivek Wadhwa's house in full tuxedo, I hate tuxedos, does anyone here know David Cheriton? Because you know for sure he does not like tuxedos and he was in a tuxedo. But I went up to him and we complained about our attire. Then I said, hey what's happening at Stanford? And he said, well there is this project called backrub, and it's search. It's page search by page rank and relevancy. Today everyone says pagerank and relevancy is obvious. Back in 1998 that was not obvious, that engineers were designing a product based on this thing called pagerank. All it was was a simple algorithm that said if a lot of people go to that website and other websites direct them there, there must be something good happening on that website. That was the original algorithm. The motivation was relevance. So I said to David, I have to meet these people. He said, you can't meet them until they're ready. Which was the following May funny enough, I waited, I called them every month for five months. And finally got my audition with Larry and Sergey. Right away they were very strategic. They said, they'll let you invest if you can get Sequoia, we don't know Sequoia but they are investors in Yahoo and because we are late to market, but we want to know we have a deal with Yahoo. So I earned my way into the investment in Google.

Marc Andreessen: I will tell one on the other side, which is Airbnb, which we were not early investors in, Airbnb is a growth round, we did the first big growth round under Airbnb at about a billion dollar valuation in 2011. And I think that will turn out to be, I believe that will turn out to be one of the spectacular growths of all time. I think this will really be a great company, so I will tell that story because it is not a story of pure genius.

We passed. I don't think we even met with them the first time around, or maybe one of our junior
Lecture 9: How to Raise Money

people did. I said earlier that venture capital is entirely a game of outliers. One of the key things of outliers is that their ideas often seem outright nuts up front. So of course having a website where you can have other people stay in your house, if you made a list of the ideas that were the most nuts that would be right there at the top. Well the second most stupid idea you can think of is having a website where you can stay at other people's houses. Airbnb deeply combines both of those bad ideas. So of course it turns out, they have unlocked an entirely new way to sell real estate, they have unlocked this gigantic global phenomenon. So part was just coming to the fact that we just whiffed on our initial analysis of the idea and the numbers were clearly proving that we were wrong. And the customer behavior was clearly proven that we were wrong.

So one of the philosophies in our firm is we are multistage, a big reason for that is so we can fix our mistakes and we can pay up to get in later when we screw up early on. The other thing I will highlight on, the other reason we pulled the trigger at a high valuation when we did was because we had spent time at that point with the founders, with Bryan, with Joe, and with Nate. And there is a friend of mine who has a great line, he says when people progress in their careers they get bigger and bigger jobs, and at some point they get the really big job. Some of the people grow into the job, and half the people swell into it. And you can kind of tell the difference. There is a point when people just lose their minds. One of the issues with these companies that are super successful and hyper growth companies, Airbnb was sort of the classic case with these super young founders that haven't ran anything before. How are they going to be at running this giant global operation? We were just tremendously impressed and are today every time we deal with all three of those guys; how mature they are, how much that are progressing. It's like they get more and more mature, they get better and better judgement, and they get more and more humble as they grow. So that made us feel really good, that not just was this business going to grow, but that these were guys who were going to be able to build something and be able to run it in a really good way.

Ron Conway: You know, people always ask me, why do you think Airbnb is going to be such a great company? Its funny, we are obsessing over Airbnb. It's because all three founders are as good as the other founder. That is very rare. In the case of Google, two founders, one of them is a little better than the other... hey, he is the CEO. Every company has a CEO. Why am I saying this? When you start a company, you have to go find somebody as good or better than you to be the co-founder. If you do that, your chance of success grow astronomically. And that is why Airbnb became so successful, so quickly. The anomaly is Mark Zuckerberg at Facebook. Yes he has an awesome team, but the Mark Zuckerberg phenomenon where is it one person, is the outlier. So when you start a company, you have got to find phenomenal co-founders.

Sam Altman: Okay, audience members.

Q: So obviously the conventionalism of why you raise money is you need it. But the more I get off conventionalism the more I hear another story on why you raise money. I am actually hearing founders say it is more to facilitate the big entity. Or in the worse case the apple hair instead of fizzling out into nothing. To what extent is that accurate thinking?

Ron Conway: Well if you pick good investors who have good rolodexes and good domain expertise in what your company does, they are going to add a lot more value than the money. Those are the types of investors you should be looking for.

Marc Andreessen: So the answer to the question is yes but in a sense it doesn't matter. Because you can not plan these things according to the downside. I mean that is the scenario that you are obviously not hoping for. While the answer is yes, that should probably not play into the decision making process too much, it might enter into which investor to raise money from, it probably doesn't enter into the whether...
to raise money question that much.

Q: If you intend to start a business that is capital equipment intensive, do you guys have any advice on how to deal with demotivation? So not everything starts in software, viral, or anything else? What should founders do for capital equipment intensive companies?

Marc Andreessen: I would double down on my previous comments on the onion theory of risk and the staging of risk and cash. Which is the more capital has the business, the more intense you have to be about exactly what is going to be required to make a business work and what the staging of milestones and risks are. In that case you want to line up, you want to be very precise on lining up, because the risk is so high that it will all go sideways, right? You want to be very precise on what you can accomplish with your A round and what is going to be a successful execution of your A round. If you raise too much money in your A round that will seriously screw you up, right, later on down the road. Because you are going to raise a C seed then the accumulative dilution will get to be too much. So you have to be precise on every single round, you have to raise as close to the exact amount of money as possible. Then you have to be as pure and clean and precise with the investors as you can possibly be about the risks and the milestones.

This, by the way, is a big thing. I am really glad you asked the question. It kind of goes back to what Parker said. If you walk into our firm, and you have Twitter or Pinterest or you have something, and it's just viral growth and it's just on fire and it's just going to go, those are the easy ones. It's like, let's just put money in it. Let's just feed the beast. But if you walk in and you're like, I got this really great idea but it's going to take three hundred million dollars staged out over the next five years probably across five rounds. It has a potential of very big outcome, but this is not Twitter. We will still do those but the operational excellence on the part of the team matters a lot more. One of the ways you convey the operational excellence is in the quality of the plan. Back to the Steve Martin thing, be so good that they can't ignore you. The plan should be precise.

Ron Conway: If you are capital equipment intensive, there are ways of borrowing money, in addition to venture capital.

Marc Andreessen: You can kick in venture debt and then later on lease financing, but then again that underlies the need for operational excellence. Because if you are going to raise debt then you really need to be precise on how you are running the company because very easy to trip the convenence on a loan and it's very easy to lose the company. So it's a thread the needle process. The demands are just a more advanced level of management than the next Snapchat.

Q: What is a sign you should not work with an investor?

Ron Conway: Well it is the inverse of what I said about a good investor. If it is an investor that has no domain expertise in your company, does not have a rolodex where they can help you with introductions both for business development and in helping you do the intros for Series A, you should not take that person's money. Especially if they are in it just to make money. And you can sus those people out pretty quickly.

Marc Andreessen: I'm glad you asked this question because it brings up a larger point. If your company is successful, at least the ones we want to invest in want to build big franchise companies, so we are talking about a ten or fifteen or twenty year journey. Ten, fifteen, or twenty years you may notice is longer than the average American marriage. This is significant. The choice of key investors, of particular investors who are going to be on the board for a company, I think is just as important as who you get married to, which is extremely important. These are people you are going to be living with, partnering

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
with, relying on, and dealing with in positions, in conditions of great stress and anxiety for a long period of time.

This is the big argument I always make, and I make it all the time, sometimes people believe it, sometimes they don't. If everything goes great, it doesn't matter who your investors are. But almost never does everything always go great. Even the big successful companies, even Facebook, all these big companies that are now considered to be very successful, all along the way all kinds of shit hit the fan over, and over, and over, and over again. There are any number of stressful board meetings and discussions, and late-night meetings with the future of the company at stake where everyone really needs to be on the same team and have the same goals, be pulling at the same direction, have a shared understanding, have the right kind of ethics, and the right kind of staying power to actually weather the storms that come up.

And one of the things you will find is a big difference for first time founders versus second time founders is almost always the second time founders take that point much more seriously after they have been through it once. So it really, really, really matters who your partner is. It really is like getting married, and it really is worth putting the same amount, maybe not the quite as much time and effort as picking your spouse, but it is worth spending significant time really understanding who you are about to be partnered with. That is way more important than did I get another five million in the valuation order, or another two million dollars in the check.

Ron Conway: I know at SV Angel, our attitude is if you invest in an entrepreneur, we are investing for life. Because if we made the right decision, we are going to invest in every company they start. Once an entrepreneur, always an entrepreneur. We actually do consider it a marriage.

Parker Conrad: I always look for, in that first meeting, do you feel like you respect this person and do you feel like you have a lot to learn from this person? Because sometimes when you meet with VCs, you feel like they are slow on the uptake, they just don't get it. Sometimes you walk in and they have just this incredible amount of insight into your business that you walk out of their like, man even if these guys didn't invest that sort of hour that I spent with them was such a great use of my time, I felt like I came out with a much clearer picture of what I need to do and where I need to go. And that is such a great microcosm of what the next couple years are going to be like. If you feel like you would really want this person to be really involved in the company, even if they didn't have a checkbook that they brought with them, that is probably a really good sign. And if not, that is probably a really bad sign.

Q: What is the constraint on how many companies you guys have invested in? Time, money, or lack of good companies?

Ron Conway: SV Angel has kind of got comfortable with one a week. You certainly can’t do more that that, and that is a staff of thirteen. So it's really the number of companies.

Marc Andreessen: Ron if you had twice the number of hours, would you invest in twice the number of companies?

Ron Conway: I would advise against that. I would rather just add more value to the existing companies. SV Angel does have a written conflict policy. But when we do end up with a conflict, it is usually because one company has morphed into another space. We don’t normally invest in companies that have a direct conflict. If we do, we disclose it to the other company, to both companies, and keep in mind at our stage, we don’t know the company’s product strategy anyway. We probably don’t know enough to disclose, but our conflict policy also talks about this really important word which is trust. In other words, we are off to a bad start if we don’t trust each other. With SV Angel, the relationship between
Lecture 9: How to Raise Money

Marc Andreessen: Let me go back to the original question. This is the question that is talked about most in our firm. So the main constraint on a top-tier venture capital firm is the concept of opportunity cost. It's the concept that means everything you do, there are a whole bunch of other things that you can't do. It's not so much the cost of we invest five million dollars into the company, the company goes wrong and we lose the money. That's not really the loss that we are worried about because the theory is we'll have the winners that will make up for that.

The cost we worry about is, every investment we make has two implications for how we run the firm. Number one rules on conflicts. So our policy for sure on venture and growth rounds is that we don't invest in conflicting companies. We only invest in one company in a category. So if we invest in MySpace, and Facebook comes along a year later, we are out. We can't do it. Every investment we make locks us out of a category. And the nature, that is a very complicated topic when we're discussing these things internally. You only know the companies that already exist, you don't know the companies that haven't been founded yet. And god help you if you invested in an early company that was not going to be the winner and you were locked out by the time the winner came along.

The other issue is opportunity cost on the time and bandwidth of the general partners. Going back to the concept of adding value, we are a general firm, we have official partners. A partner can maybe be on ten to twelve boards in total if they are completely, fully loaded. Basically you want to think of it as a ticket that you have a limited number of holes you can punch, every time you make an investment you punch a hole. When you are out of holes to punch, you are done, you can't make any new investments. That is very much how venture capital operates. A way to look at it is, every board slot that our GP's have at any given time is an asset to the firm. They can be deployed against an opportunity. But every time we make an investment, it takes the number of slots we can punch down by one. So it reduces the ability for the firm to do new deals. Every investment we make forecloses not just the competitive set but other deals that we will simply run out of time. This goes back to what I said earlier. This company is fairly good, it seems fairly obvious that it's going to get venture funding, why didn't you fund it? Well, on its own, if we had unlimited capacity, we probably would have, but relative to getting blocked by a competitive set, and not having that board seat for an even better opportunity, we pass on that basis a lot.

Q: What would convince you to invest in a company with no product to show. What do you make that judgement based on?

Ron Conway: What would convince us, is what usually convinces us, is the founder and their team themselves. So we invest in people first, not necessarily the product idea. The product idea tends to morph a lot. So we will invest in the team first. If it's pre users, the valuation is going to tend to be correspondingly lower unless one of the founders has a success track record.

Marc Andreessen: For us, if there's nothing at the time of investment it's always almost, other than a plan, it's always, usually a founder we have worked with before. Or a founder that is very well known. You know, in these conversations, the default assumption is that we're all starting consumer web companies or consumer mobile companies. You know there are other company categories. For example, enterprise software companies, SaaS, application companies. It's much more common that there is no MVP, it's much more common that it's a cold start. And it is much more common that they build a product in the A round. There is no point in having an MVP, the customer isn’t going to buy an MVP, the customer actually needs a full product when they first start using it. So the company actually needs to raise five or ten million dollars to get the first product built. But in almost all those cases that is going to be a founder that has done it before.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwin.nl 06-42804208)
Q: What is the ideal board structure?

Parker Conrad: We are fortunate to have, there’s myself and my co-founder and a partner from Andreessen Horowitz. I think it removes the fear, creates a little more trust. It removes the fear that someone is going to come in and fire you arbitrarily because it's time for a big company CEO. But in most cases if you trust the people you are working with, it shouldn't really be an issue. Things almost never come to a board vote, and by the time they do it's something is deeply broken at that point any way. Most of the power that VCs have is outside of the board structure. There are protective covenants that are built into the financing round, you can't take on debt, you can't sell the company, there are certain things you can't do without them agreeing to it anyway.

It's less of a big deal than people make it out to be. What I found, as a founder, if things are going well at the company, you have unlimited power vis a vis your investors. No matter what the board structure is, no matter the convenants in the round. If you say, this is what I want to do, I think this is what we need to do. Even if it's a good investor, or a bad investor, they are like, let's make it happen. They want to ride this rocketship with you, and when things go badly it doesn't matter what protections you have built into the system for yourself.

Marc Andreessen: When a company gets in dire straights, it doesn't matter what the terms for the prior round are, they all get renegotiated. I have been on boards for twenty years, I have never been in an important board vote that mattered. It's never been a vote, many discussions, many controversies, never a vote. The decision has always been clear by the end. It’s almost always unanimous. And so I think the decision is almost always around the intangibles and not around the details.

Sam Altman: Okay thank you guys for coming today!
Lecture 9: Why Software Is Eating The World


Marc Andreessen

This week, Hewlett-Packard (where I am on the board) announced that it is exploring jettisoning its struggling PC business in favor of investing more heavily in software, where it sees better potential for growth. Meanwhile, Google plans to buy up the cellphone handset maker Motorola Mobility. Both moves surprised the tech world. But both moves are also in line with a trend I’ve observed, one that makes me optimistic about the future growth of the American and world economies, despite the recent turmoil in the stock market.

In an interview with WSJ’s Kevin Delaney, Groupon and LinkedIn investor Marc Andreessen insists that the recent popularity of tech companies does not constitute a bubble. He also stressed that both Apple and Google are undervalued and that “the market doesn’t like tech.”

In short, software is eating the world.

More than 10 years after the peak of the 1990s dot-com bubble, a dozen or so new Internet companies like Facebook and Twitter are sparking controversy in Silicon Valley, due to their rapidly growing private market valuations, and even the occasional successful IPO. With scars from the heyday of Webvan and Pets.com still fresh in the investor psyche, people are asking, “Isn’t this just a dangerous new bubble?”

I, along with others, have been arguing the other side of the case. (I am co-founder and general partner of venture capital firm Andreessen-Horowitz, which has invested in Facebook, Groupon, Skype, Twitter, Zynga, and Foursquare, among others. I am also personally an investor in LinkedIn.) We believe that many of the prominent new Internet companies are building real, high-growth, high-margin, highly defensible businesses.

Today’s stock market actually hates technology, as shown by all-time low price/earnings ratios for major public technology companies. Apple, for example, has a P/E ratio of around 15.2—about the same as the broader stock market, despite Apple’s immense profitability and dominant market position (Apple in the last couple weeks became the biggest company in America, judged by market capitalization, surpassing Exxon Mobil). And, perhaps most telling, you can’t have a bubble when people are constantly screaming ‘Bubble!’
Lecture 9: How to Convince Investors

http://paulgraham.com/convince.html

Paul Graham

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 x?")

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, and they'll sense that. You don't have to be confident and mistaken. Formidable is roughly justifiably confident.

But if it is, you'll be telling the truth when you tell investors it's 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 convincing 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 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.
Lecture 9: How to raise money

http://paulgraham.com/fr.html

Paul Graham

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 worst 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 they 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 inexperienced at raising money. Inexperience there doesn't mean they are unattractive. Being a noob at technology would, if all investors have is worthless initially. And while most investors 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. [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 x $15k x 18 = $5.4 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 bedesperate. 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.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
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.

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 term sheet. 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 term sheet. 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 schleps 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 its 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 you 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 Griffel, 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 Tomarello for reading drafts of this.
Lecture 9: Linkedin’s Series B Picture to Greylock by Reid Hoffman

http://reidhoffman.org/linkedin-pitch-to-greylock/

Reid Hoffman

At Greylock, my partners and I are driven by one guiding mission: always help entrepreneurs. It doesn’t matter whether an entrepreneur is in our portfolio, whether we’re considering an investment, or whether we’re casually meeting for the first time. Entrepreneurs often ask me for help with their pitch decks. Because we value integrity and confidentiality at Greylock, we never share an entrepreneur’s pitch deck with others. What I’ve honorably been able to do, however, is share the deck I used to pitch LinkedIn to Greylock for a Series B investment back in 2004.

This past May was the 10th anniversary of LinkedIn, and while reflecting on my entrepreneurial journey, I realized that no one gets to see the presentation decks for successful companies. This gave me an idea: I could help many more entrepreneurs by making the deck available not just to the Greylock network of entrepreneurs, but to everyone.

Today, I share the Series B deck with you, too. It has many stylistic errors — and a few substantive ones, too — that I would now change having learned more, but I realized that it still provides useful insights for entrepreneurs and startup participants outside of the Greylock network, particularly across three areas of interest:

- how entrepreneurs should approach the pitch process
- the evolution of LinkedIn as a company
- the consumer internet landscape in 2004 vs. today

—Reid Hoffman

MySpace after raising its premium round from Benchmark and Kleiner in the fall of 2003. Facebook, by the way, was not yet on most people’s radars in the summer of 2004.

Friendster’s valuation set the tone for the entire social networking space. Friendster and MySpace had millions of users, a ton of engagement, and all the press attention they wanted. Press and analysts characterized LinkedIn in one of two ways: “LinkedIn is an interesting niche that might be worth paying attention to” or “LinkedIn is the Friendster for business”. Neither is a particularly good backdrop for trying to raise capital, because

1. we weren’t the natural leader of a market or technology trend that everyone was paying attention to,
2. we didn’t have substantial organic growth, and
3. we had no revenue.

Advice

Investors see a lot of pitches. In a single year, the classic general partner in a venture firm is exposed to around 5,000 pitches; decides to look more closely at 600 to 800 of them; and ends up doing between 0 and 2 deals. The goal of an entrepreneur is to be one of those deals.

First, understand your audience. Research prospective investors thoroughly. What kinds of businesses are they looking at? What model/criteria/triggers do they use to judge whether a project will be successful or not? If you don’t have some sense of their points of view, your likelihood of making the pitch go well is more random. You may happen to emphasize the right points that pique an investor’s interest, but you shouldn’t leave your financing up to chance.

Second, understand the broader financing climate. In 2004, investors regained interest in the consumer internet again. Friendster raised a big round in 2003; MySpace started gaining traction. But with so many investors still licking their wounds from the dot-com bust, many focused on proven business models, such as advertising or e-commerce. As a result, we knew that our pitch would need to steer into investors’ biggest concern: the lack of revenue.

Context

In 2004, the consumer internet was just beginning to rebound. Friendster was at its height, strongly battling
What is LinkedIn? The graphic we chose emphasizes that it is a network of people.

Why is it valuable? Because you can find and contact people you need.

How is this different? Because unlike Google search or other means, it involves people you already trust.

Although we knew that the recruiting space would be our initial business opportunity, we believed then — and know now — that LinkedIn is more than just a recruiting business. Thus, our pitch framed LinkedIn as a platform for finding the people you need, which we called "professional people search 2.0", making the parallel to Google because investors understood that Google was valuable. (On slide 5, I begin explaining the importance of pitching by analogy.)

If we framed LinkedIn as only a "jobs/classifieds" website, most smart venture capitalists would not have invested because that seemed to lack the potential to be a broad platform that could sustain a large business. Ultimately, Greylock's investment thesis was that LinkedIn would be a great recruiting business with an option for more.

Advice

Open with your investment thesis, what prospective investors must believe in order to want to be shareholders of your company. Your first slide should articulate the investment thesis in generally 3 to 8 bullet points. Then, spend the rest of the pitch backing up those claims and increasing investors' confidence in your investment thesis.

For example, if I were pitching LinkedIn's Series B today with what I now know about successful pitches, the investment thesis would be:

1. Massively valuable properties will be built off networks.
2. There will be different networks for different domains.
3. The professional domain will be one massively valuable network.
4. We are the leader in the professional domain with viral growth.
5. Great businesses can be built off this network, starting with matching talent and opportunity.
6. It's a network effects business, which means it has inherent defensibility with a network.

Clearly articulate your investment thesis so investors can offer feedback that helps you refine it, eventually getting to a place where you both agree on it. Any disagreement will likely cause serious problems down the road.

Context

Normally, you’d expect us to explain our product — i.e., what “professional people search 2.0" is. Instead, our strategy was to steer immediately into the revenue question because that was the top concern of investors in 2004. And

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
We made the mistake of listing three different revenue streams. As it happened, we did end up pursuing all three lines of business. And LinkedIn proved to be an exception to the rule of thumb: our diverse business lines have been a strong plus. General rules sometimes have important exceptions which can be tremendously valuable. That’s true in business strategy, entrepreneurship, and even pitch advice.

**Context**

With the revenue question out of the way, we were ready to explain our product. We had two questions to answer: What is the product? And why is it new?

We argued that the way professional people search was done at the time (“1.0”) was inadequate. To make this argument, we listed three important professional business problems (finding service providers, finding job candidates, and reaching professionals) that were time-consuming and difficult to accomplish with existing technologies.

The key problem with existing technologies was adverse selection, specifically concerning the incentives of participants and the reputation systems:

- In the yellow pages, people wanted to be found but the way they represented themselves had nothing to do with how good they were.
- In old-school resume databases, most talented professionals didn’t want or need to participate.
- In directories, professionals only wanted to participate if other talented professionals did, too.

So, how do you create a platform where talented professionals can participate, be found, and be contactable? Our answer: a network. A network solves this problem because all of their friends and contacts would be on it — and friends of their friends. Creating the right incentives and reputation system would lead to a directory people would be a part of.

**Advice**

Steer into your investors’ objections. There will be one to three issues that are potentially problematic for your financing — address them head on. You have the most attention from investors in the first couple slides. Most investors arrive with questions, and if you proactively show you understand their principal concerns, you earn their attention for the rest of your pitch.

For consumer internet properties in 2004, because we had just gone through the dot-com winter, investors’ principal concern was whether or not you could make money. As you recall, we began our pitch by steering into the revenue question because we didn’t have tens of millions of users or a growth curve off the chart; otherwise, we would’ve started with one of those.

In 2013, it’s whether you can break through the noise. Today, there are probably a thousand consumer internet startups founded every quarter — how do you become one of the 1 to 3 that matter in a 7-year timeframe? Those are the kinds of objections you need to steer into at the beginning of your pitch.
thesis is worthwhile, rather than having the entrepreneur tell them what to conclude. For early stage companies, it’s important to show that you’re on path, that you have prospects, and that you can get to your vision.

**Context**

Once investors believed that professional people search was valuable, the next question was whether internet 2.0 (the move to networks) amplified that value considerably. To demonstrate this, we showed how the 2.0 transformation created value in other markets.

First, we looked at goods listings. 1.0 is businesses like online classifieds for newspapers, which were unsuccessful. eBay, on the other hand, was really valuable. What’s the difference with eBay? eBay has a network. It has reputation. It has transactional histories. Adding a network to online classifieds made it valuable. (Just think of how valuable Craigslist would be if it had identity and reputation.)

**Advice**

**Pitch by analogy.** Every great consumer internet company grows up to be a unique organization. But in the early days, you want to use analogies to successful outcomes to describe what your company is and what its potential could be. Time is short — it helps to refer to what those investors already understand.

The best pitch I know was in Hollywood for a film called *Man’s Best Friend*. The pitch was “*Jaws with Paws*”. Investors thought that if the movie *Jaws* was a huge success, maybe a similar premise on land with a dog could be a huge success. The movie turned out to be terrible, but the pitch was excellent.

**Context**

Next, we looked at online payments. Although online payment transfer already existed with banks, PayPal’s model of a network of payments is what made it unique. The problem we chose for this example was fraud. It is difficult for banks to detect fraud because they don’t have access to the whole payments network; they only have access to individual nodes in that network.

Another reason we used PayPal as an example was to remind investors that I was part of PayPal’s founding team — a minor example of showing, not telling.

**Advice**

Understand where analogies apply and where they do not. Pitch by analogy but don’t necessarily reason by analogy. Reasoning by analogy, when you’re developing your business strategy, is dangerous.

In startup land, you’re running across a minefield, so the details matter and you have to be careful with your analogies as you conceive strategy. In fact, when I’m the investor listening to a pitch, one detail I consider is whether the entrepreneur is being too deluded by their analogies and not thinking hard enough about exception cases.

**Context**

In our third example, we contrasted Altavista and its search algorithm versus Google and its search algorithm PageRank. PageRank is one more use of networks: search results that leverage an overall network of pages rather than just rely on the occurrences of terms.

**Advice**

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Context

Finally, we contrasted LinkedIn against both Monster and LexisNexis because we wanted to show that LinkedIn would add value to all valuable applications related to professional people search — e.g., recruiting (represented by Monster) and service directories (represented by LexisNexis).

How valuable could LinkedIn be? Well, we point to Monster, LexisNexis, and other information service providers and say, “Whatever multiplier you applied to the last 3 slides — about how networks amplify the value of companies like eBay and PayPal — apply that now to LinkedIn.”

LinkedIn would create a networked resume document — a resume 2.0 — instead of traditional job listings with private resumes. When you’re finding people on LinkedIn, you’re finding them through a network as opposed to a resume database. We also knew that networks would improve information reputation systems, allowing people to find the best possible information.

Today, networks underlie the information reputation systems of many consumer internet companies, including LinkedIn, Facebook, and Twitter.

Advice

Avoid debating the validity of your analogies. If someone pushes back and tries to challenge elements of an analogy, don’t let yourself get drawn into a back and forth. Analogies are a conceptual framework, so they’re not going to be 100% accurate.

However, so many entrepreneurs try to pitch by analogy that some investors have fatigue when they see it. If you have a good analogy, use it. But if you don’t have a good one, don’t include one just to have one. It’s better to have no analogy than a bad one.

Context

Here, we remind investors that this investment decision comes down to whether or not they believe that a network creates huge value.

Even though we knew we would monetize, we argue that investors shouldn’t be thinking about our current revenue numbers. Instead, they should think about the network we established because that’s what ultimately wins. We spent our Series A capital building the network, so we needed
Lecture 9: How to Raise Money

Context

In slides 10 and 11, we compared what we promised in our Series A pitch with what we actually did. Our overdelivery against our Series A predictions provided strong evidence to new investors that we could execute against our plan.

LinkedIn’s Series B was a concept pitch because our data at that point wasn’t impressive. At the time, Friendster had about 10.5 MM users and MySpace had 2.5 MM users. LinkedIn, at the time, was still approaching its first million users and did not have a dime in revenue.

Advice

Your investment thesis is either concept-driven or data-driven. Which kind you are pitching?

In a data pitch, you lead with the data because you are emphasizing how good the data already is. Investors therefore evaluate your company based on the data. When LinkedIn went public, it was a data pitch to public market investors. We showed investors a multi-year track record of data.

If it’s a concept pitch, on the other hand, there may be data, but the data supports a yet undeveloped concept. A concept pitch shows your vision for how the future will be and how you will get to that future, so investors will want to buy a piece of it. Thus, concept pitches depend more on promised future data rather than present data. When you’re doing a concept pitch, it’s especially important to consider pitching by analogy.

Context

In slides 12 and 13, we asserted three things:

- The professional space is valuable, even if you think it’s less valuable than other spaces.
- LinkedIn can provide the best product and be the strong market leader in this space.
One ingredient this pitch lacks, which I now think is essential to modern pitches, is our risk factors. Experienced investors know there are always risks. If they ask you about your risk factors and you can’t answer, you’ve lost all credibility because they assume you are either dishonest or dumb.

Dishonest if you’ve thought about the risk factors but choose not to share them, which is a bad way to build trust and a partnership. Dumb if you aren’t smart enough to understand that all projects have risk factors — including yours.

Explicitly identify the risks that could thwart your success and how you will mitigate them. And instead of waiting until investors ask about your risks, share them proactively so you build trust.

Context

Our investors didn’t need to worry about the top-left quadrant of competition (Friendster, Myspace, Orkut, and Tribe.net) because those focused on the social side, which we asserted was an entirely different space from LinkedIn’s professional space. Our investors didn’t need to worry about BranchIT, Visible Path, etc., because those mostly focused on enterprises instead of individual professionals.

Instead, what investors should have paid attention to was the companies in the blue box — Ryze, OpenBC, ZeroDegrees, and Spoke. If investors agreed, their next question was usually “What’s your competitive strategy in each case?” We thought we had a viable competitive strategy in each of these cases.

Advice

Entrepreneurs often say they have no competition, assuming that’s an impressive claim. But if you claim that you don’t have competition, you either believe the market is completely inefficient or no one else thinks your space is valuable. Both are folly.

The market is efficient, eventually — if a valuable opportunity emerges, others will discover it. To build credibility with investors, you want to show that you understand the competitive risks and show why you’re going to win.

Express your competitive advantage. Why are you going to break out of the pack? What is your advantage? An understanding of product-market fit? Is it a technology advantage? What’s your differential business strategy? Your differential growth strategy? Your differential product? If you aren’t clear and decisive, investors won’t believe you have an edge that can lead to success.

Growth Includes Major Groups
Providing Strong Brand Endorsement

Context

This slide is a testimony to our nervousness about our organic growth compared to Friendster and MySpace. Again, because this pitch is a concept pitch, we wanted to convince investors to bet on our future. Thus, we wanted to show successful organizations that were committed to our success.

Advice

This is mostly a mistake slide because customer slides are more appropriate for enterprise pitches. Great customers are predictive of future customers for enterprise businesses. On the consumer internet, however, this is a sign of trouble because it indicates that the entrepreneur may not understand how the consumer internet works.

Generally speaking, consumer internet businesses need grassroots and individual adoption rather than organizations promoting it. Although we knew this at the time, we violated it because we were nervous about our adoption and because we thought we might be a unique exception.
Lecture 9: How to Raise Money

Context

To bring home the argument that investors should believe in us in the future, we show what we accomplished after spending the majority of our Series A $4.7MM financing.

Advice

As I said earlier, you want to show focus in your decks by emphasizing what you're really betting on. However, show some maneuverability. Don't just say that you have five different options. Instead, say that you're doing one, but you also have some fall-back or maneuvering options.

For example, if we were doing the Series B pitch in 2004 with my knowledge of today, we would emphasize that LinkedIn would start by transforming one business — the recruiting industry, by shifting it from a posting model to a searching model. Then, in our talking points, we would highlight some of the other businesses we can transform with our platform.

"Invest in A, but here’s B to show that we could contain that risk." Investors would appreciate this because you're identifying a reasonable risk and demonstrating that you have actually thought about what you would do if the primary plan doesn’t play out as you expect.

Advice

It's always better to have less slides, but it's much more important to have a great deck. A great deck needs to address all important concerns and tell your story effectively. Sometimes, that means setting up a narrative over several slides.

Don't stress about the exact number of slides. Entrepreneurs often hear advice that their decks should be a particular length. I, for example, recommend a length of 20 to 25 slides. But these are only rules of thumb, which means you can violate them if you have a good reason.

LinkedIn's Series B deck contained a couple slides that we spent little to no time talking about, but we included them because they had information that showed investors we had thought about all the important details. Even though we glossed over those slides, investors knew they could come back to those slides later and dig into them if necessary.

Context

By this point in the presentation, our goal was to lead investors to believe:

1. A network-empowered people search application could be really valuable.
2. We are the market leader in establishing the network.
3. We have a viable plan for establishing revenue off the network.

Here, we argue that we have a viral product that creates a network that possesses network effects. We didn’t try to establish the viral dynamic; we just asserted it. I explained more of the strategy in person, but we also didn’t want to
cover it much in writing because virality was a pretty deep secret in the industry at the time.

Today, given that virality has achieved buzzword status, you usually have to show evidence that you know what virality is and how it works. Surprisingly, there are still few people who understand virality.

Advice

When pitching VCs, think about the individual partner in the context of their partnership.

Make sure the individual partner has strong cohesion and trust with the partnership and is responsible for your type of business.

In the financing process, the individual venture partner performs the due diligence on the substance behind a company’s pitch — e.g., the investment thesis, the competition, the people, etc. The partner then asserts the investment thesis and why they believe it to the partnership.

Ultimately, you’re selling the partnership, so give the individual partner the talking points to be successful. What will that partner tell their partners? Put yourself in their shoes.

The Network Enables Revenue

Value of the network | Revenue model (high-level)
--- | ---
Network-based reputation system creates trust key to high transaction volume | eBay takes 1% of transaction
Reputation system is not directly monetized

Network-based fraud detection system makes the business economically viable | PayPal takes 1% of transaction
Fraud detection is not directly monetized

Network-based page rank system produces best search results | AdWords are shown in search results
Page rank system is not directly monetized

... and of course, user-based network effects are key to making the business valuable

Context

Earlier in the pitch, we argued that establishing the network was our first priority with our Series A. Our Series B, however, was about getting to revenue, so how were we going to do that? Before we detail our revenue plans, we remind investors why the value of the network created good businesses across eBay, PayPal, and Google — our previous examples of network-enabled 2.0 businesses.

Each example generates considerable revenue out of the network despite not charging people to be in the network. Each example has a network powering its revenue-generating applications. You don’t pay for the eBay reputation system; you pay for the transaction. You don’t pay for PayPal’s fraud system, but the fraud system enables those transactions to go through or not and to be profitable. You pay for AdWords, not the regular search results.

Advice

People frequently think the most fundamental strategy of a startup is its product strategy. In fact, the most fundamental strategy is the financing strategy. If your company runs out of gas (finance), your company will die no matter how good your product strategy is. Frequently, the product/service strategy is harder to develop, but the financing strategy should be there first.

LinkedIn’s financing played out as follows: The Series A was a concept pitch for building the network. The Series B was a concept pitch for getting to revenue. The Series C had to be a data pitch that showed either how we could get to profitability or that we were profitable. (In fact, our Series C showed profitability and so we focused on growth.) And Series D ended up being “We can scale to a big opportunity.”

The Network Enables Revenue

Table:

<table>
<thead>
<tr>
<th>Value of the network</th>
<th>Revenue model (in detail)</th>
</tr>
</thead>
</table>
| Network-based reputation system creates trust key to high transaction volume | 1. InLeads: contextual search text ads
2. Opportunities: network-enabled job listings
3. Network Plus: network-enabled subscription
4. Reputation system is not directly monetized (Universal service stays free for all users) |

Building the network is the challenge, and LinkedIn has the foundation

Context

To this point, our conceptual argument for getting to revenue was this: Just as networks enabled revenue across eBay, PayPal, and Google in the previous slide, the network will enable revenue for LinkedIn, too. But we also had a concrete plan, which we began detailing in this slide.

We return to the three revenue models from the beginning of the pitch, introducing the product names for the first time: an ads product called InLeads, a job listings product called Opportunities, and a subscriptions product called Network Plus.

Again, our argument is that we have a network that’s valuable which will enable revenue. This slide reinforces the fact that we achieved a valuable outcome in series A (building the network), so it made sense not to have revenue yet. But now it was time to get to revenue.

Advice

Always think about the next round. The usual tempo for raising money from venture capital is at a minimum of a year between financings. Every time you raise a round, you
should be thinking about the next round. Who will be the next investors you pitch? What will their concerns be? What will you need to solve next? How will you raise money later?

Expect that future investors will look at today’s deck. When I created our Series A deck, I presented a growth curve that would be good enough to get an investment, but I also had confidence that I could beat it. I wanted to be able to go into my Series B presentation and say, “Here’s what I said before, and here’s how I did.” Because we beat our Series A expectations for network growth, investors could comfortably trust our promise to build revenue with our Series B financing.

**Context**

Above is another version of saying that we built a network and it’s time to build the revenue.

To some degree, this is somewhat an empty placeholder slide, but as the last slide in the “network enables revenue” sequence, it provides a clear visual for understanding the concept. I’m not sure I would keep this slide in retrospect; however, I remember that we talked to it well.

**Advice**

Reinforce key concepts when delivering a concept pitch. Diagrams are one way to accomplish this, helping investors visualize key concepts. In our pitch, we wanted to make sure investors understood that you build the network first and then you can build a platform of businesses on top.

It’s helpful (but not mandatory) to put your thesis in each of the titles. If an investor sequenced through the titles, they’d be able to get a sense of the flow of the argument. This is especially helpful when investors are sharing the decks with their investment partners.

**Context**

Investors frequently ask, “What’s an existing business that’s like your business that you could occupy?” This is why many entrepreneurs include a slide about their Total Addressable Market (TAM), the underlying revenue opportunity for a product or service.

Smart venture capitalists saw that we had interesting comparables, but didn’t spend much time on this slide. The
Lecture 9: How to Raise Money

Context

Above is a mockup of how InLeads, a product that wasn’t built yet, could work. The point was to demonstrate why people would pay for a listing on LinkedIn as opposed to just buying Adwords.

We only used InLeads as a brand in this deck. By the time we got around to building our Marketing Solutions business, we had a different concept of advertising than InLeads.

Advice

Show your product rather than saying you intend to build a best-of-breed product. Ideally, you want to have the product built. Otherwise, you should show what you have in mind with a mockup.

A mockup is better than nothing because it increases investors’ confidence by showing that you’re thinking concretely about the product and allows them to evaluate your plan.

1. Market for InLeads: Search Advertising is Large and Growing Fast

Online advertising market ($B/Year)

- $10,000
- $7,000
- $4,000
- $1,000
- $0

<table>
<thead>
<tr>
<th>2000</th>
<th>2001</th>
<th>2002</th>
<th>2003</th>
<th>2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>$6,007</td>
<td>$7,184</td>
<td>$5,098</td>
<td>$7,067</td>
<td>$5,995</td>
</tr>
</tbody>
</table>

LinkedIn InLeads Improves on Google AdWords model

- 5X daily professional reach
- 10X daily member page views

"You closed over $100K worth of business in 105 days and currently have $500K of deals in the pipeline thanks to LinkedIn/InLeads. Communications"

Advice

When in doubt, lead with what will make the most sense to investors. We never launched InLeads, but we led with it because in 2004 everyone understood that AdWords was a golden goose. Since we weren’t decided on the exact sequencing of our revenue plan, we led in the pitch with the one whose value proposition could easily be understood by investors.

2. LinkedIn Opportunities Lets Users Leverage Their Network to Hire

Context

The above mockup reinforces the idea that a network overlay improves job postings and reference checking. For example, the network enables job seekers to get introductions to people in the group they’re being hired into, or to people who could talk to that hiring group, or to the hiring manager themself. The network enables job seekers to reference check whether a company is a good place to work, or find someone to perform an informational interview with.

Fast forward to 2013 and most people recognize that LinkedIn has a pool of professional identities/CVs/profiles, but they don’t fully recognize how deep the reference checking capabilities of our product is. You can use the information on LinkedIn to identify referential information, allowing you to prioritize whom you’re reaching out to. You can reference check a wide variety of professionals — including domain experts and service providers — both before and after you meet them. Plus, it goes in both directions — employees can reference check prospective managers, and managers can reference check prospective employees.

The difference between a great, a good, a mediocre and a bad hire are enormously disastrous in term of potential impact on your business. Improving the quality of hires or increasing the speed of the hiring process saves professionals time and money. That’s a worthy value proposition.

Advice

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
In concept pitches, you’re selling a story, so naturally there will be people who don’t believe that story. That’s okay because you don’t need everyone to believe it; you only need the right people to believe it. Naturally, you want everyone to think your business is amazing, but don’t get deluded by that. Startup financing is not a popularity contest; everyone saying yes is irrelevant to you. It’s more important to have the right person say yes than it is to have everyone say yes.

The best outcome is an investor who can help you build the company and realize a market opportunity. Put another way, the ideal financing partner is a financing cofounder. This is why already-wealthy entrepreneurs raise money from experienced investors for their next startup: they know partnering with angels and venture capitalists is about more than just the money.

Sadly, many investors actually add negative value, so an investor who adds no value (“dumb money”) but who doesn’t interfere with the operational process can sometimes be a decent outcome. But ideally you find an investor who can proactively add value (“smart money”).

How do you know if an investor will add value? Pay attention to whether they are being constructive during the financing process. Do they understand your market? Are their questions the same questions that keep you up at night? Are you learning from their feedback? Are they passionate about the problem you’re trying to solve?

Although using quotes wasn’t particularly helpful in our pitch, it is important to always be talking with smart people to solicit their feedback. Talk to your network to evaluate your ideas and evaluate your pitch. Half the time, the feedback is irrelevant — even smart investors may not understand your idea — but you should still be listening carefully because you may learn valuable insights. If you are speaking with a bunch of smart people and a similar thread emerges, there’s something to that thread and you should pay attention to it.

Be wary of confirmation bias. It’s only natural that an entrepreneur wants to hear that their idea is great, but you don’t want people telling you that because it doesn’t help you. The questions you should always ask are: What’s wrong? What’s broken? Why won’t it work? What do you think the risks are? People by default will want to give you good feedback rather than bad, so you have to ask negative questions.

How do you know if an investor will add value? Pay attention to whether they are being constructive during the financing process. Do they understand your market? Are their questions the same questions that keep you up at night? Are you learning from their feedback? Are they passionate about the problem you’re trying to solve?

Context

This slide is overkill, and in hindsight I would delete it. We were trying to show rather than tell investors that our job listings product would have customers by displaying customer feedback. We used quotes as evidence of product-market fit, showing that there were people who would actually use the value propositions of our network-based listings. But this was a weak effort. If investors didn’t already believe this, this slide did not help. And if they do believe, this slide did not increment their belief.

Advice

Internal data is preferable over anecdotal third party data. Since we didn’t have data to back up various assertions about our product-market fit, we had to rely on quotes from customers and press sources, which weakened our pitch.

Although using quotes wasn’t particularly helpful in our pitch, it is important to always be talking with smart people to solicit their feedback. Talk to your network to evaluate your ideas and evaluate your pitch. Half the time, the feedback is irrelevant — even smart investors may not understand your idea — but you should still be listening carefully because you may learn valuable insights. If you are speaking with a bunch of smart people and a similar thread emerges, there’s something to that thread and you should pay attention to it.

Be wary of confirmation bias. It’s only natural that an entrepreneur wants to hear that their idea is great, but you don’t want people telling you that because it doesn’t help you. The questions you should always ask are: What’s wrong? What’s broken? Why won’t it work? What do you think the risks are? People by default will want to give you good feedback rather than bad, so you have to ask negative questions.

2. Market for Opportunities: Existing Online Job Markets Are Large But Inefficient

<table>
<thead>
<tr>
<th>Revenue (2004: $M)</th>
<th>Average Unique Monthly Visitors ($)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monster</td>
<td>515</td>
</tr>
<tr>
<td>Careerbuilder</td>
<td>175</td>
</tr>
<tr>
<td>Yahoo HotJobs</td>
<td>94</td>
</tr>
<tr>
<td>LinkedIn</td>
<td>n/a</td>
</tr>
<tr>
<td>Total Market</td>
<td>827</td>
</tr>
</tbody>
</table>

Employers will increasingly tap LinkedIn’s rich profiles to find passive job candidates rather than pay Monster to access its resume database.

Context

As a platform for changing the world of work and enabling individuals, jobs is one important application built upon LinkedIn’s professional network — but it’s not the only one. We believed the general discovery of people would be really/valuable, which is why we say “LinkedIn is not only about jobs” in the slide above.

The purpose of this slide was to show that we were focused on the recruiting industry as a key market, but we were nervous about being pigeonholed as a job listings site. Among Silicon Valley investors, Monster and HotJobs were not considered great, investable businesses — even though Monster had a $4B market cap — because of the constant amount of churn for jobs listings. What’s more, the sales and marketing costs of acquiring listings are expensive.

Advice

Take competition against your potential revenue streams seriously. Being detailed about your competition, especially listing the specific companies, helps increase investor confidence.
**Identify the right metrics for success.** Focus on revenue and activity rather than market cap or Total Addressable Market numbers. Although we had some real growth in a month, we had no revenue, so our numbers were not yet impressive. However, by highlighting revenue and growth as the most important success metrics in the recruiting space, we proved that we were serious and thoughtful about the space and showed that we understood the metrics that matter.

**Context**

Above is the mockup for the subscriptions product, then called Network Plus. Rather than just saying we would have a product that does X, we wanted to show a specific product idea, backed by market research, that was tangible in detail. Again we used quotes to show folks who were already finding this value proposition useful.

What we also showed in this slide was that even though every LinkedIn member had the ability to search, there was a difference between free and paid accounts. We were underpromising and overdelivered, ultimately giving paid accounts visibility of the entire network. Today, LinkedIn Premium subscriptions provide more powerful tools — such as contacting anyone with InMail Messages and the full list of who's viewed your profile — to easily find, contact, and manage the right people for their professional needs.

Designing the subscription product this way created more value for our members, which made the network itself more valuable. The number one value for the entire history of LinkedIn has been “Members first.” Even while building the subscription product, it was important to have a product that worked for subscribers and members.

**Advice**

Underpromise and overdeliver. Internally, our team expected that paid members would potentially get access to the whole network, but we had to make sure that the network would be comfortable with this. So we showed four degrees to our investors, to be sure.

**Show that you’re paying attention to the market.** Instead of merely saying that we knew product-market fit is key, we wanted to show that we did the work. We used quotes to show that we were talking to credible individuals struggling to solve our problem who were giving us feedback on our products. There are other methods, of course, such as graphs and data. But if we’d simply said, “We’re focused on product-market fit,” that doesn’t show, that tells.

**Context**

Because Network Plus was going to be a subscription service, we were worried that an investor might point out that this was not how Monster’s job listings worked at the time. As a comparable, we point to the market for personal dating — which has a similar value proposition to professional networking — because it uses subscription services and has large dollar spend.

Subscriptions was one of the major business models we pitched, but in 2004, few investors believed subscriptions would work. Most investors saw the internet as an advertising medium, even today. LinkedIn is one of the companies that has proven that subscriptions can work for consumer internet products.

**Advice**

One of the virtues that entrepreneurs get from talking to many investors during the financing process is a wisdom of crowds that helps you figure out what the real risks are. When I was pitching SocialNet (my first startup company which was a dating service similar to Match.com), what I heard from numerous investors was this concern: “If someone uses SocialNet to find a partner and succeeds, then they’re no longer a customer, which means you intrinsically have a massive churn problem.” As it turns out, that signal was absolutely right. Listing businesses have to solve this churn problem by figuring out how to have lifelong customers, which is always a target for great businesses.
Again, we address revenue to build confidence that a business will come. We did listings first because it shows all members the value of the network. Next, we did subscriptions with a focus on HR/recruiting/jobs because we realized that was what people primarily used our search for. We also figured out that subscriptions would be the market entry strategy for our recruiting product.

LinkedIn has such a valuable, aspirational user base that the subscriptions product isn’t just popular with jobhunters, it’s popular with business developers, journalists, venture capitalists, networkers, and more.

As I said earlier, we never launched InLeads. What we discovered while thinking through the search ads product was that people aren’t looking for people who are advertising to them; they’re looking for the people they’re trying to find. So we only got to doing self-service advertising much later. In the meantime, we reserved a space with brand advertising.

**Advice**

**Be decisive and ship.** Including specific dates, for example, shows decisiveness. Being decisive doesn’t mean that you have to stick with your decisions. Good investors expect you to iterate often as you figure out what product will grow your market. Greylock didn’t mind that we didn’t follow this timeline closely. What mattered was that we made progress, while being definitive about our company’s playbook and demonstrating an ability to make difficult decisions.

While it’s important to think carefully about your future, don’t think too far into the future. You will change, the world will change, and the competitive landscape will change. It is useful, however, to have a strategic direction supported by confident projections. After all, successful businesses outdo their competition by better anticipating long-term strategy.
assumptions can later be validated by due diligence. You don’t want investors to fixate on claims that appear crazy.

Context

Sometimes, in Series B pitches, a slide like this would be an appendix slide. As a matter of fact, for the vast majority of our presentations, we didn’t spend any time on this slide. Credible investors were glad to see it because the slide showed that we were thinking about the revenue question.

Again, we were nervous about getting the financing past the lack of revenue so we kept shooting at it from different directions. Here we’re saying, “Look, with a Series B we can get to operating profitability,” which, by the way, we did. In fact, when LinkedIn received the HBS Award in 2010 for Entrepreneurial Company of the Year, our Series A investor Mark Kvamme showed that with a one-year delay (which we deliberately did to grow the network) we closely followed the revenue model we gave in our Series A. That almost never happens with Series A projections.

Advice

A successful financing process obviously results in you raising capital for your company, but it also results in a partnership that delivers benefits beyond just money. For example, great investors can significantly boost the strength of your network, which helps in recruiting employees and acquiring customers. Great investors can also be a source of network intelligence, so you can better prepare for likely challenges and opportunities ahead.

Business and Tech Team Has Strong Track-Record

Reid Hoffman, CEO
• Former EVP and Founding Board Member, PayPal
• Investor in Friendster, Inteport, Blu Apron, Vendr etc

Sarah Imbach, Chief of Staff and VP
• Former SVP, Operations, PayPal

Alen Blazic, VP of Product
• Former Director of Product Design, Socialist.com

Jean-Luc Vaillant, VP of Engineering
• Former Director of Engineering and Software Development, Logitech, Spytite

Konstantin Guercke, VP of Marketing
• Former VP of Marketing, Presenter, Blazovin

Eric Ly, CTO
• Founder and former CTO, Nethosphere (sold to Critical Path)

Matt Cohler, Director of Corporate Development
• Former top-ranked analyst at Morgary, managing at AsiaInfo, published in HBR

Technical team with experience at Apple, Cisco, Hewlitt, Citigroup, Sprout, TIBCO, VERITAS, etc

Advice

You have the most attention from investors in the first 60 seconds of your pitch, so how you begin is incredibly important. One common mistake is putting the team slide early in the deck. The team behind your idea is critical, but don’t open with that. Instead, open with the investment thesis.

This advice applies to seed funding rounds, too. Yes, seed investors understand that early stage companies have many unknowns and the idea will change a lot, so they look carefully at the people to see whether the team will be able to adapt. But even at this stage, lead with your overall investment thesis. Persuade investors that your investment thesis is intriguing, then show who can make it happen.

Board and Other Investors Are Sequoia and Top-Tier Angels

Reid Hoffman
• CEO of LinkedIn
• Former EVP and Founding Board Member, PayPal, leading angel investor

Mark Kvamme
• Partner, Sequoia Capital
• Former CEO and Chairman, CKS Group

Bob DeStatis
• Former CMO and Co-Founder, Aruba

Additional angel investors (not on Board of Directors)
• Peter Thiel, Co-Founder and Former CEO, PayPal
• Marc Andreessen, Co-Founder and Former CTO, Netscape
• Joe Kraus, Co-Founder and Former SVP Marketing, Excite
• Andrew Amsec, Former CTO, Wired and Former Partner, August Capital
• Gil Penchina, VP International, eBay

Context

Because we had zero revenue at the time of the pitch, it was especially important for us to give our investors confidence that our team would execute. To this end, we made sure to include both a thorough slide of our team and a thorough slide of the broader network of investors and advisors backing the company.

One benefit of going with top-tier venture capital is that they tend to have good prediction track records, which helps establish credibility for early stage businesses. Given that Sequoia Capital is a top-tier VC, we gave Sequoia

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208) 220 / 412
prominent placement (including their logo) on this slide. As a partner at Greylock, I've encouraged my portfolio companies to do the same with Greylock.

**Advice**

People frequently say that you should work your way up to the investors you most want to work with. Instead, identify the investors you most want and the investors who are most likely to say yes. These are not necessarily the same people (though it’s excellent if they are). Approach your likely investors and your ideal investors at the same time, because your likely investors provide a temporal forcing function by which you might end up with your ideal investors.

Although I had identified David Sze and Greylock as the investors I most wanted, I pitched another firm first. They responded by rolling in with an offer before I pitched Greylock. Since David and the partnership at Greylock knew I had an offer, they gave me a term sheet the day after I pitched them.

Stay aboveboard so you keep trust with prospective investors. I was forthright with the first venture firm about my intentions, telling them that they received the first look, and that I would talk to a few more firms to put time pressure. And Greylock knew I had gotten a term sheet to accelerate their decision process.

Naturally, the first firm was disappointed when I turned them down, but they weren’t surprised. While they worried their early offer would be shopped, sometimes firms end up getting the deal that way, or sometimes they get to participate in the deal because of their early offer. Investors know that smart entrepreneurs intend to get a good term sheet early, but they have to try to win the deal.

**Context**

As we’re wrapping up the Series B pitch, we wanted to remind investors that we spent the $4 million from our Series A and accomplished a lot, to increase their confidence in giving us $10 million to turn the next card.

**Advice**

When you’re starting out, you want to show investors that you can build into a great business, which sometimes leads to hyperbolic statements. *It’s okay to be hyperbolic, to be aggressive, to be visionary.* Avoid adverbs and adjectives when possible, but you want to show that you have strong forward motion. In this pitch, we wanted to show that we believed this would be a huge, valuable platform. In the end, what matters to investors is that the investment turns out well.

**Find and Contact the People You Need Through the People You Already Trust**

**Context**

The reason we reused this slide from the beginning of the presentation was to indicate the end of the presentation, while returning to the high line of how to conceptualize the business and reminding investors of the value proposition. We left this slide up on the screen while taking questions, so it served as a buffer for the appendix.

The reason we didn’t keep an “Appendix” slide up was that we didn’t want investors to see that we had an appendix, requiring us to go through all our prepared slides. We were happy to go to the appendix, but we only wanted to get to it if we were asked the pertinent questions.

**Advice**

You should end on a slide that you want people to be paying attention to. A placeholder slide that says only “Appendix” or “Q&A” is never that. Instead, close with your investment thesis. We should have blended this slide with the previous slide (#36) to end on one investment thesis slide, which would be left up on the screen while answering questions. You want that slide to remind your investors why they should invest in your company.

I now believe you should begin and end with the investment thesis. The beginning is when you have the most attention, and the end is when you should return to the most fundamental topic to discuss with your investors. Plus, as you talk about the investment thesis, you learn from your investors how to improve it. Smart investors should add to your thinking about the investment thesis.

[http://startupclass.samaltman.com/](http://startupclass.samaltman.com/) (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
APPENDIX

Context

Our presentation included five appendix slides, which we've omitted for brevity.

1. A better way to find professionals online
   This slide showed a mock-up of what a professional search product could look like.

2. Professional networking, not social networking
   This slide addressed the question: “Isn’t social much more valuable than professional?” We showed how the two domains work differently. This was particularly prescient when Facebook came along. Social versus professional as categories took a long time to establish, but even then with Friendster and others, the two domains work differently.

3. Key assumptions behind financials
   We showed this slide to address the assumptions behind our financial models going to profitability.

4. LinkedIn InLeads improves on Google AdWords model
   This slide was in case investors focused on advertising, since we had led the presentation with that and many investors assumed that consumer internet companies required advertising models. Fortunately, we almost never used the slide because smart investors like Greylock picked up that there was an HR space.

5. Reputation-based prioritization of candidates
   This slide showed how LinkedIn’s Opportunities model would improve the traditional job site model.

Advice

You don’t have to have an appendix, but if you anticipate serious questions from the kinds of investors you want: preparing appendix slides with structured answers is impressive, showing that you’ve considered all of your business’ challenges, opportunities, and comparisons. Appendix content typically fall under two categories: providing additional information or addressing objections.

Closing Reflections

Today, LinkedIn seems like an obvious investment to have made in 2004. But at the time of the Series B financing, LinkedIn had spent its $4 million from Series A building a network that was much smaller than Friendster, MySpace, etc. We had no revenue, or even revenue-capable products.

The journey from founding, to multiple rounds of financing, to IPO, was not easy. All startups go through real Valley-of-

---

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Lecture 9: Ron Conway's Recommended Reading List

https://docs.google.com/spreadsheets/d/1KLCdlp4SW7Z64W_DWGwwuTlfH1GXRAMtP xv4EB5gKu0/edit?pli=1
<table>
<thead>
<tr>
<th>Book</th>
<th>Author</th>
<th>Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Product Design for the Web</em></td>
<td><strong>Randy J. Hunt</strong>&lt;br&gt;Creative Director @ Etsy</td>
<td>Web designers are no longer just web designers. To create a successful web product that's as large as Etsy, Facebook, Twitter, or Pinterest--or even as small as a tiny app--you need to know more than just HTML and CSS. You need to understand how to create meaningful online experiences so that users want to come back again and again.</td>
</tr>
<tr>
<td><em>Talking To Humans</em></td>
<td><strong>Frank Rimalovski &amp; Giff Constable</strong>&lt;br&gt;Frank Rimalovski - Executive Director, NYU Entrepreneurial Institute @ NYU&lt;br&gt;Giff Constable - CEO, Neo</td>
<td>Explains how to do the customer development interviews in a way that will get you the most accurate and actionable feedback.</td>
</tr>
<tr>
<td><em>Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers</em></td>
<td><strong>Geoffrey A. Moore</strong>&lt;br&gt;Organizational theorist, management consultant and author</td>
<td>Bestselling guide that created a new game plan for marketing in high-tech industries.</td>
</tr>
<tr>
<td><em>The Four Steps to the Eppiphany</em></td>
<td><strong>Steve Blank</strong>&lt;br&gt;Silicon Valley serial-entrepreneur and academician who is recognized for developing the Customer Development methodology, which launched the Lean Startup movement.</td>
<td>Offers the practical and proven four-step Customer Development process for search and offers insight into what makes some startups successful and leaves some startups unsuccessful.</td>
</tr>
<tr>
<td><em>Founders at Work: Stories of Startups' Early Days</em></td>
<td><strong>Jessica Livingston</strong>&lt;br&gt;Partner at YCombinator</td>
<td>Collection of interviews with founders of famous technology companies about what happened in the very earliest days.</td>
</tr>
<tr>
<td><em>Thinking, Fast and Slow</em></td>
<td><strong>Daniel Kahneman</strong>&lt;br&gt;Israeli-American psychologist. He shared the 2002 Nobel Memorial Prize in Economic Sciences with Vernon L. Smith. Kahneman is notable for his work on the psychology of judgment and decision-making, behavioral economics and hedonic psychology.</td>
<td>Summarizes research that he conducted over decades, often in collaboration with Amos Tversky. It covers all three phases of his career: his early days working oncognitive bias, his work on prospect theory, and his later work on happiness.</td>
</tr>
<tr>
<td><em>The Score Takes Care of Itself: My Philosophy of Leadership</em></td>
<td><strong>Bill Walsh with Steve Jamison &amp; Craig Walsh</strong>&lt;br&gt;Ex-49er's head coach</td>
<td>Words of wisdown from Bill Walsh to inspire, inform, and enlighten leaders in all professions.</td>
</tr>
<tr>
<td><em>Gates of Fire: An Epic Novel of the Battle of Thermopylae</em></td>
<td><strong>Steven Pressfield</strong>&lt;br&gt;American author, of historical fiction and non-fiction, and screenplays</td>
<td>The story of the battle of 300 Spartan soldiers against the Persian army.</td>
</tr>
<tr>
<td><em>High Output Management</em></td>
<td><strong>Andrew S. Grove</strong>&lt;br&gt;Former COO, Chairman and CEO, currently senior advisor at Intel Corporation</td>
<td>User-friendly guide to the art and science of management.</td>
</tr>
<tr>
<td><em>The Idea Factory: Bell Labs and the Great Age of American Innovation</em></td>
<td><strong>Jon Gertner</strong>&lt;br&gt;Editor at Fast Company Magazine</td>
<td>The definitive history of America's greatest incubator of technological innovation.</td>
</tr>
<tr>
<td><em>Onward</em></td>
<td><strong>Howard Schultz</strong>&lt;br&gt;Chairman &amp; CEO @ Starbucks</td>
<td>Schultz's story of his return to Starbucks, and the success of the company in a tumultuous economic time in history.</td>
</tr>
</tbody>
</table>

---

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
<table>
<thead>
<tr>
<th>Book</th>
<th>Author</th>
<th>Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Art Spirit</td>
<td>Robert Henri</td>
<td>Notes, articles, fragments of letters and talks to students, bearing on the concept and technique of picture making, the study of art generally, and on appreciation</td>
</tr>
<tr>
<td>How to Raise Money</td>
<td>Paul Graham</td>
<td>Advice on raising money for young startups.</td>
</tr>
<tr>
<td>StartUp School</td>
<td>YCombinator</td>
<td>Startup School is a free to attend, one day conference where you'll hear stories and practical advice from founders and investors.</td>
</tr>
<tr>
<td>Sequoia - Writing a Business Plan</td>
<td>Sequoia Capital</td>
<td>Information on successful business plan structure.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Website</th>
<th>Author</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>TechCrunch</td>
<td>Multiple</td>
<td>Latest technology news and information.</td>
</tr>
<tr>
<td>PandoDaily</td>
<td>Sarah Lacy</td>
<td>To be the site-of-record for that startup root-system and everything that springs up from it, cycle-after-cycle.</td>
</tr>
<tr>
<td>VentureBeat</td>
<td>Multiple</td>
<td>Leading source for news &amp; perspective on tech innovation.</td>
</tr>
<tr>
<td>Hacker News</td>
<td>YCombinator</td>
<td>Newsfeed of current trending articles in tech.</td>
</tr>
<tr>
<td>Ben Horowitz Blog</td>
<td>Ben Horowitz</td>
<td>Reflect on the career experiences &amp; lessons learned—as computer science student, software engineer, cofounder, CEO, fund raiser, company acquirer and seller.</td>
</tr>
<tr>
<td>Fred Wilson's Blog</td>
<td>Fred Wilson</td>
<td>Daily thoughts and conversation.</td>
</tr>
<tr>
<td>HeavyBit Library</td>
<td>Multiple. Program for funded companies building developer-facing products and services</td>
<td>Videos and talks.</td>
</tr>
</tbody>
</table>
10. Company culture & building a team I
Lecture 10: Company Culture and Building a Team, Part I

http://startupclass.samaltman.com/courses/lec10/

Alfred Lin

I'll set the stage with some slides and a few comments but the main stage is going to be with Brian when he comes up and talks about how he built the AirBnB culture. So, you are here, you have been following the presentations, so you know how to get started. You have built a team, you know how to build your product, it's off the ground, it's growing. People love it, you figured out how to do that. You figured out to create a special one of a kind company with monopoly powers, that's big. And the market you are chasing after is slightly bigger than the paper airplane business, so you are good, right? So now what?

So we are here to submit that actually culture is the thing that is actually going to be very, very important to scale the business as well as your team. And hopefully, after this talk you will be able to know: What is culture? Why does it matter? How do you create your core values? And think about elements that fit together for core values and culture that create a high performance team. Get some best practices for the culture.

What is culture? Anybody want to take a guess at how one should define this? {Simple values in a team?} Yeah, that's good. Did you look that up because you had a computer and an internet connection? These are some definitions that you will find in Webster's dictionary but we are Stanford. This is kind of a trick question. It's a CS class, the questions are never straight forward.

The real question is, what is the company culture going to be. Culture, we can generally talk about society, about groups, about places, or things. Here we're talking about company culture. So how does one define company culture? We can take the previous definition and modify it a little bit. This is a hint of how we may want to define company culture. Every day blank and blank of each member of the team in pursuit of our company blank. Some people have filled these in with different things. The first blank, A, could be assumptions, beliefs, values, now my favorite is core values. The second blank for the B blank, people said behavior, my favorite is action. How do you act? In pursuit of goals, that's kind of weak, in pursuit of big and hairy audacious goals is a little stronger, but a better definition is in the pursuit of mission.

So now that we have that definition, what do we do with that and why does it matter? This is a quote from Gandhi “Your beliefs become your thoughts. Your thoughts become your words. Your words become your actions. Your actions become the habits. Your habits become your values. And your values become your destiny.” If you don't have a good culture in the company you can't pursue your destiny.

Why it matters is that it becomes the first principles you sort of go back to when you make decisions. It becomes a way to align people on values that matter to the company. It provides a certain level of stability to fall back on. And it provides level of trust, people sort of trust each other with, but it also gives us a list with which you should be able to figure out what to do and what not to do. And what the more important thing about that is what not to do. Then finally the other thing that is important is it allows you to retain the right employees. There are people in this world that are not going to be a fit for your company, but if you have good strong culture, and the strong core values, you'll know who you want to retain and who you truly do not want to retain. And if you take the first letter of those it happens to help you move faster.
Another reason, you're thinking that's like all mushy stuff, this is actually more scientific stuff. So here are indices from 1994 to 2003, stock market indices of companies in the S&P500, and the Russell three thousand, and the Fortune 100 Best Companies to Work For. All these companies out there and they picked out companies that they believed were the best companies to work for. The stock market returns of those companies happen to be 11.8%, which is almost twice that of the other two indices. And so there is real power in companies that treat their employees well, where there's a lot of trust, there is a lot of strong culture.

So how do you create a set of values and define the culture etc? Get asked that a lot. You have to start with the leader of the company and the founder, and ask yourself what are the values that are the most important to you? Of those things, that are most important in the business? Who are the types of people you like working with? And what are their values? And through that you distill together what a set of values are. Think about all the people that you've never liked working with. What values do they have? Think of the opposite of that. Maybe those should be considered values for your company. Finally remember the values have to support your mission and if they don't support your mission, you're missing something. Then the last final checks are they have to be creditable, they have to be uniquely tied to match your mission. So at Zappos, in terms of uniquely applied to the mission, we were focused on creating a culture that was going to provide great customer service. So the first core value we had, was to deliver wow through service. We are very specific that we wanted to deliver great customer service and it was going to be a wow experience. And then below that we wanted to serve.

I had a paragraph talking about what we mean by that, we wanted to support them, doing the wow through service, and support people such as our employees, our customers, our brand partners, and investors. On terms of the opposite thing, we generally didn't like working with arrogant people, so one of our core values at Zappos was to be humble. So those are two examples where we create core values in a way that became credible and uniquely tied to our mission.

So you go through this process, you come up with a few core values, these might be some of them, whether it's honesty, integrity, service, teamwork. It might be a list, you might start with three, might end up with a list of ten, you might have a list of thirty. It's a good start. When Zappos went through this process we asked all the employees at the time what core values they can identify with and they came up with thirty-seven. We whittled that down to about ten. And it took a year to do this, that's a long time and you might want to ask why. Well if you just come up with the word honesty, give me a break, everybody wants the culture to be honest, nobody is going to say I want to be lied to every day. Service, what do you mean by service? There's got to be a lot more depth in this than that. And everybody talks about teamwork, but there's a difference in level of teamwork that you see in an intramural sports team vs a baseball team. How do you dive deeper into teamwork? What are the things that don't work for a team? A lot of it has to do with communication, a lot has to do with things that people of study, and you may one go deeper into that. At Zappos we thought about, well there are a lot of smart people in this room. When they're fighting with each other and trying to figure out who's right and who's not, it's probably not the best use of time. We wanted everyone to build off each other and help each other make any idea better. The result is that the company gets a better idea, not that any individual person is right. So we wanted to make and still, this idea that it's company first, then your department, then your the team, then yourself. And how do you do that?

We are going to go a level deeper in that. There's another great element of high performing teams that I really like. Which is this pyramid that was created by Patrick Lencioni, and he wrote this book, “The Five Dysfunctions of a Team.” And the reason this is interesting is he talks about the breakdowns of the team. A lot of teams break down because they have no trust and even if you had trust, why do you need trust? If you have trust, you can actually have debates and conflict and get to the right answer. If you
don't have conflicts and debate, it's the blind leading the blind. How do you actually know you got to the right answer before you commit to something? So people are not actually wanting to commit, they're afraid of committing.

Let's say you get to the next level when you are actually able to commit. What goes wrong then? It's usually because people are not held accountable to things that they committed to. And if people are not held accountable to the things that they committed to, then they can't get results. If you think about the company as a black box and results, whether it's financial, whether you produce a great product, or anything like that as the output, one of the major inputs is the culture of the company. Some other best practices, we are actually going to talk though in the Q and A because I think they are going to blend into the conversation, is that you want to incorporate your mission into your values, we talked about that.

Performance, you need to think a bit harder, deeper, longer about your values than you might initially think you need to. One of the things I think a lot of companies don't actually do is, they interview for technical fit or skill fit, a competency in that realm, but they don't actually interview for the culture fit, whether someone will actually believe in and follow the mission. I think that is a big, big no no. I think you can have the smartest engineer in the world but if they don't believe the mission they are not going to pour their heart and soul into it. And that's one of the things where if you actually start thinking about culture, from the interview process, to performance reviews, to making sure that's a daily habit, you get a lot further with making a great culture.

One final point made here, culture, just like customer service or fitness, is like motherhood and apple pie. Everybody wants to provide great customer service, every company wants to have great culture. What they fail to do is make it a daily habit. You just can't be fit, if you don't do it as a daily habit. Eventually you get out of shape, then you get fat, and then you say, Oh I have to go on a crash diet to get back into shape. That doesn't quite work, and the same is true with something like culture. So I think we checked all of these off, so we can go into Q and A with Brian.

Brian Chesky: Hello everybody. It's quiet in here, I'll be honest, now I feel a little less on edge. Nothing worse than a room full of people really, really quiet staring at you, but now I feel better.

Alfred Lin: Well I did it for five or ten minutes, you can do it a little longer. So Brian, could you talk about the process by which you came to understand that culture was important to AirBnB and building a company?

Brian Chesky: Yeah, so I think one of the things we realized is, to just give you, I won't tell the full story of how Airbnb came to be. Some of you may know it. So here's the very short version the story, Airbnb wasn't meant to be the company we were trying to start. I quit my job, I was living in LA. One day I drove to San Francisco, became roommates with my friend from college, from the Rhode Island School of Design, Joe Gebbia, and I had one thousand dollars in the bank and the rent was one thousand one hundred and fifty dollars. So that weekend this international design convention was coming to San Francisco, all the hotels were sold out, so we decided to turn the house into a bed and breakfast for the conference. I didn't have a bed, Joe had three air beds, we pulled them out of the closet and called it The Air Bed and Breakfast. That's how the company started.

I recall the story ten thousand times by the way, some version of that story, and I didn't think I'd ever tell that a second time. I remember growing up, I also went to college, my parents were social workers and never thought about me going to art school. They had worried that maybe I would not get a job after college, which I'm sure most parents worry about. Make sure you promise me you get a job with
health insurance, I end up starting Airbnb. I remember her telling me, I guess you never got the job with health insurance.

The reason I say this though is, Airbnb was never meant to be the big idea. It was meant to be the thing to pay the rent so we could think of the big idea. Along the way, by solving our problem it became the big idea. So alongside that, we won’t touch on how we built a product, that’s probably another conversation that’s being talked about, you have to build a team and a great company. And in the early days, we had three co-founders, Joe, Nate, and myself. I think of one of the reasons we’re successful was that I was really lucky. I don’t think it was really lucky to pick up the idea Airbnb and I don’t think we were lucky to be successful once we had a team. I think we could have come up with a lot of other ideas and been just as successful. I think I was lucky in that I found two great people that I wanted to start up a business with, people I admired. That almost intimidated me by how smart they were. I think that's what the first thing is, to build a team that is so talented that they kind of, slightly make you uncomfortable to be with them, because you know you are going to have to raise your game to be with them.

And then when we were working together in the early days, this is 2008, the first thing is we were like a family. You think about founders like parents and the company like a child. That child will manifest in many ways, behaviors that parents have in the relationship. If the parents are functional but not working together then the child is, frankly, going to be pretty fucked up. You don’t want that. You want your culture to be awesome. And so Joe, Nate, and I were a total family the beginning, we usually worked eighteen hours a day, seven days a week. I remember when we were at Y Combinator, we worked together, we ate food together, or even went to the gym together. We may as well have gotten jumpsuit, we didn’t go that far. It was like we were a mission, a special force.

We had this amazing shared way of doing things with amazing accountability, and that was the DNA of a company. And then we were thinking, at some point you build the product to phase two, which is building the company that builds the product. So a lot of the talk is about how to build a product, how to get product market fit. Once people do that, now you have to build a company. It doesn't matter how great your original product idea is, if you cannot build a great company then your product will not endure. As we thought about this, we realized we wanted to build a company for the long term. The last thing I want is to build something, think about it this way, if your company is like a child, a parent wants their child to outlive him, or her. It would be a tragedy for us to outlive our company. To just watch it rise and fall. We didn't want that.

We wanted a company that would endure. To do that, we started noticing companies have something in common. Companies around for a really long time had a clear mission. A clear sense of values, and they had a shared way of doing something that was unique to them and was really special. And so Joe, Nate, and I, when we were three people, decided to look around at companies. I noticed Apple, Steve Jobs' core value was that he believed people with passion could change the world. He said our products change but our value never has. We learned about Amazon, we learned about Nike, we learned about companies in the early days. You can even use this to talk about nations. Even a nation has core values and a declaration so that the nation may endure longer. We started to realize that we needed to have intention, culture needs to be designed. And that is how we got connected. Because we were funded by Sequoia, Alfred just joined them with Zappos, and I was told Zappos had an amazing culture. We went to Las Vegas and met up with Tony and we learned about it.

Alfred Lin: So what did you learn?

Brian Chesky: Well, you guys are crazy. The thing we learned, and we were three people, was if culture is a way of doing things, there really are two arts. One is behaviors and those can change maybe fifty
years from now. There will be rituals and behaviors that change, be different. But there have to be some things that never change. Some principles, some ideas that endure, that make you, you. And I think of core values, integrity, honesty, those aren't core values. Those are values that everyone should have. But there have to be like three, five, six things that are unique to you. And you can probably think about this in your life. What is different about you, that every single other person, if you could only tell them three or four things, you would want them to know about you. And we realized that when Zappos was one hundred employees, they wrote down these ten core values. The thing I learned from Tony is I wish I didn't wait until I was one hundred employees to write down a core value. I was talking to Sam, he thinks we are the only company to write our core values down before we hired anyone.

Alfred Lin: How long did it take you to hire your first employee?

Brian Chesky: So the first employee was our first engineer, I think we looked for him for four or five months. I probably looked through thousands of people and interviewed hundreds of people.

Alfred Lin: By then, when you hired him, when did you write it? Was it on day one or two or was it three?

Brian Chesky: I think we started working on it around the time of Y Combinator, which would've been January 2009. It was probably a process that evolved over the course of six to seven months. We finished Y Combinator in April 2009, hired our first engineer in July something like that. Probably six months. Some people ask why did you spend so much time on hiring your first engineer. I think bringing in your first engineer is like bringing in a DNA chip to the company. This person, if we're successful, there were going to be a thousand people just like him or her in that company, it still wasn't a matter of getting somebody to build the next three features we need the ship for users. There was something much more long-term and much more enduring which was, do I want to work with one hundred thousand more people like this? Now, you want diversity to play, you want diversity of background, age. You don't want diversity of values, you want very homogenous beliefs. That's the one thing that shouldn't be diverse.

Alfred Lin: So what were these values?

Brian Chesky: Six core values, I'll talk about maybe three of them. So the first core value we talk about is champion in mission. And what it really means is that we want to hire people that are here for a mission. We don't want people here because they think we have a great valuation, they like our office design, they need a job, or they think it's hot. We want people to be here for the one thing that will never change, and that's our mission. And just to tell you a quick story about our mission, Airbnb, a lot of people describe it as a way to book a room or book a house and travel around the world. And that's what we do, but that is not why we do it. To answer the question on what our mission is, is to tell you a story I think it describes it.

In 2012, I met a host named Sebastian, we do these new jobs around the world where we do meet ups. Sebastian is probably late fifties in north London. Sebastian looks at me and says, "Brian there is this word you never use on your website." And I say, "What's that word?" And he says, "That word is friendship. I would love to read a story about friendship." I said, "Okay read me a story about friendship." He says, "Six months ago the brunt of riots broke out in front of my home and I was very scared. The next day my mom called me to make sure I was ok, I said yeah mom I'm ok. And she goes, what about the house?" He says, "The house is OK as well." He said, "Here's the interesting thing, from the time the riots broke out to the time my mom called me was a twenty four hour window of time. In the periods between that time, seven of my previous Airbnb guests called me just to make sure I was okay." He said, "Think about that, seven of my own guests called me before my own mother did."
I think that says more about his mother than his guests. But in this summer on a typical night or a peak night, we would have twenty-five thousand people staying in homes and living together and they were coming from a hundred different countries in the world. North Korea, India, Syria, Cuba. So just hearing that story, at our core that is what we are about. That’s much more than just booking a room or traveling. That what we are about is we want to help bring the world together. We want to do that by giving a sense of belonging anywhere you go. Our mission is to belong anywhere. So five years from now, twenty years from now, maybe we’re still selling rooms and homes to meet in or not, but I can guarantee you we’re always going be about a sense of belonging and bringing people together. And that’s the more enduring idea.

So when we hire people, the first thing to make sure it is that, if that is your mission, you need to champion your mission. You champion the mission by living the mission. Do you believe in it? Do you have stories about it? Do you use the product? Do you believe in the product? I used to ask crazy questions, one of the crazy questions Sam reminds me of, I use to interview people. So I interviewed the first three hundred employees at Airbnb which people think I’m really neurotic and they may also be true. I used to ask them a question, if you had a year left to live would you take this job? I amended it, people who say yes probably don’t like their families. So I changed it to ten years. I feel like you should use whatever time you have left to live. Whatever you want to do in those last ten years you should just do. I really want you to think about that, that was enough time for you to do something you really cared about and the answer doesn't have to be this company. I say fine if what you’re meant to do is travel or start a company just do that, don’t come here. Go do that.

So there is this old parable about two men laying bricks. Somebody comes up to the first man and says what are you doing? I’m building a wall. He asks the other guy, he says I’m building a Cathedral. There’s a job and there is a calling. We want to hire people not only looking for jobs, but a calling. And that’s the first value, champion the mission.

I don’t want to take all the time, I will talk about just one more. The second value relates to being frugal and I will tell you a story. By the way, all the founding stories of your company end up becoming the things that you keep talking about to thousands of people, kind of like a child, that these things keep coming back later in life. So Airbnb, I think Marc Andreessen said in the last talk that it was the worst idea that ever worked. I remember people thinking we were crazy, I remember telling people about the idea, I actually told Paul Graham, I said we have this idea it’s called Airbnb. He asked, people are actually doing this? I said, yeah. Question was, what’s wrong with them? So I knew the interview wasn't going well and in the interview Paul Graham I think wasn't going to accept us. And we told the story of how we funded the company and here's how it goes. We're introduced, Michael Seibel, who I think is a partner of Y Combinator, introduced me and Joe to like fifteen investors in the Valley, including some of the ones that have been here and all of them said no to the company. They could have bought ten percent of the company for one hundred and fifty thousand dollars. They all said no, they thought it was crazy. No one would ever stay in someone's home.

So we ended up funding the company with credit cards and you know those binders kids in school put baseball cards in? We had to put them in those, we had to put them somewhere. That’s how many credit cards we had and we were completely in debt. And in the fall of 2008, we provided housing for the Democratic and Republican National Convention. We had this crazy idea because we weren't selling a lot of homes. In the year after we launched, we had a hundred people a day visiting our website and two bookings, which is generally bad. It’s like releasing a song and a year later three people are listing to it a day. It’s probably not going to be a very popular song. But I believed in it. Joe and Nate believed in it.
So we are completely in debt, right, and we get this idea. We are an air bed and breakfast providing housing for the Democratic Republican National Convention. What if we made a breakfast cereal for like the Democratic National Convention? And we came up with this Obama themed cereal. And we called it Obama-O’s, the breakfast of change. Then we came up with a Republican themed cereal for John McCain. We found out he was a Captain in the navy. So we came up with… Captain-McCain’s, a maverick in every bite. We had zero dollars and without any money, we tried calling General Mills they told us to stop calling them or they would get a restraining order so that didn't work. We found a local, though an alumni of RISD, who made a thousand boxes deal for us. We ended up sending them to press and eventually within a week it got on national television, national news. We made forty thousand dollars selling breakfast cereal.

In 2008 we made five thousand dollars from our website and we made forty thousand dollars selling breakfast cereal. I remember my mom asking so are you are a cereal company now. And that wasn't that bad part, the bad part was the honest answer which was technically, yes. But the reason I tell that is our second core value is to be a cereal entrepreneur. I'm sorry for the cheesy pun. But be a serial entrepreneur. We really mean, is that we believe constraints bring out the creativity. When you raise eight hundred million dollars suddenly all that scrappiness, it's easy to lose that scrappiness. It’s easier for people to tell you I just need this fifty thousand dollar contract. I need this, I need that. When people are desperate and not being a little bit frugal, not being creative, or tell me they can't do something, I'll just take a box of cereal and even the suggestion of Obama-O’s is you need to be scrappy and frugal. So again a lot of the founding DNA of your company becomes these values, these principles. Everyone knows that if you don't give a crap you shouldn't be here. And it doesn't mean you have to give a crap just means you have to do be here. You also have to be creative to be an entrepreneur and super scrappy. And these are some of the values we learned.

Alfred Lin: So you guys should start to think about questions when we open up to questions from the audience but I have a few more questions to ask. So this all sounds nice, stories are great. The people here are a pretty skeptical group. It's a CS department class, probably left brain focused. This feels like a softie, right brain focus. How does having a strong culture help you make important, tough decisions?

Brian Chesky: Well I think that having it so, here's the thing about culture. There are three things they never tell you about culture. First thing is they never tell you anything about culture. No one talks about culture and no one ever tells the need to have strong culture. So there's tons of articles about building a great product, there's tons of articles on growth and adaption, and a few things about culture. It's a mystical thing that’s soft and fuzzy. That's the first problem. The second problem is it is hard to measure. Things that are hard to measure often get discounted. These are two very hard things. The third thing, the biggest problem, it doesn't pay off in the short term. If you wanted to start up a company and sell it in one year, the one thing I would tell you to do is fuck up the culture. Just hire people quickly. Culture makes you hire really slowly, makes you deliberate about your decisions that in the near term can slow progress. Putting an investment into the company short term.

First thing is the need to be very clear about what's unique to you that you stand for. Once you do that, you need to hire people that believe in that. You need to make sure you hire and fire based on these values. One thing we do is constantly repeat over and over again when we interview, we want to make sure they are world class and fit the culture. The first thing I used to ask people at the end of the interview, I would say if you can hire, this is a functional question, if you could hire anybody in the world, would you hire the person sitting across from you? If our vision is the best in the world, why aren't we hiring the best in the world? Every single person is meant to hire a person better than the previous people. You are constantly raising the bar. Then we have separate people called core values interviewers who aren't in the function. So it you are an engineer, the core values interviewers are never
engineers because we don't want them to be biased and say well I know how good they are. And they interview just for values, to make sure that people care about the same thing.
In many ways, it's two books for the price of one.

It's a funny, wry, honest autobiography of a striving Asian American who's got the drive and ambition of his family deepy ingrained, and at the same time the irreverence of a slacker college student who's looking to create a "vibe" that makes the people around him feel like they're part of something special.

Tony's childhood is a textbook entrepreneur's trial and error. His first job, raising and selling worms isn't a huge success. But then, from the back page of Boy's Life magazine, Tony tries first greeting cards, and then selling custom buttons. The button business takes off, becoming something of a family legacy handed down from brother to brother, and making hundreds of dollars a month as the mail-order business grew.

Early on the postman was Tony's friend. And in many ways the success at Zappos today can be tracked back to that feeling that Tony had as a child--of wanting something magical to come in the mail, and enjoying the feeling of getting that mail-order burst of pleasure when the mailman delivers his anticipated gifts.

Of course, Tony isn't the only entrepreneur who started out of the back of Boy's Life, or who dabbled in magic as a way to both understand human emotions and put on a show. (Yes, I had doves, and he didn't--but the impulse is the same).

But what you begin to understand as you pore over his immensely readable book is that Zappos today is rooted in Tony's youthful pleasures and passions. In fact, Tony found a lot of inspiration in the rave scene, throwing massive parties and embracing the music, lights, and smoke that turned many individuals into a communal, and connected gathering. It's kind of hard to think of Tony as a raver, but then again, what does a "raver" look like?

Says Hsieh "It's actually really funny. I've had people come up to me and say oh, I used to go to raves. I would never have guessed that about them. So I don't know what a raver looks like 10 years later in general."

But today Zappos has an employee culture that seems very much of one mind, focused on customer service and not in some sort of cookie-cutter corporate way. Zappos really cares that you're happy, and it's baked into their beliefs, their customer interaction, and even the way they hire.

"When we hire people we do two sets of interviews. The hiring manager and his or her team will do the standard fit within the team, relevant experience, technical ability and so on. But then our HR department does a second set of interviews purely for culture fit."

"We've formalized the definition of our culture into 10 core values. Basically what we're looking for are people whose personal values match our corporate values. They're just
naturally living the brand. Wherever they are whether they're in the office or off the clock."

The Zappos Core Values are:
1. Deliver Wow Through Service
2. Embrace and Drive Change
3. Create Fun and a Little Weirdness
4. Be Adventurous, Creative and Open-Minded
5. Pursue Growth and Learning
6. Build Open and Honest Relationships with Communication
7. Build a Positive Team and Family Spirit
8. Do More with Less
9. Be Passionate and Determined
10. Be Humble

And the thing that Zappos figured out, and continues to deliver on, is the idea that people who don't fit the company culture are better off being paid to leave.

"Everyone that's hired, it doesn't matter what position—you can be an accountant, lawyer, software developer—goes through the exact same training as our call center reps. It's a four-week training program and then they're actually on the phone for two weeks taking calls from customers. At the end of that first week of training we make an offer to the entire class that we'll pay you for the time you've already spent training plus a bonus of $2,000 to quit and leave the company right now."

Paying new hires to leave may seem counter-intuitive, but for Tony, it makes simple sense. "Really, the goal of that originally was to weed out the people that are just there for a paycheck."

In the end, the culture is about more than money. "It's not me saying to our employees, this is where our culture is. It's more about giving employees permission and encouraging them to just be themselves.

For Tony, building Zappos hasn't been easy. In fact, the story has more twists and turns than you might imagine as in the early days when Sequoia Capital was ready to throw in the towel, "I think they just assumed the company would either go out of business or possibly get funding from someone else. But this was back in what 2002. So the dot-com crash had just happened, Pets.com went out of business, and e-commerce in general were not looked favorably upon. So I don't think it was Sequoia. I think VCs in general at the time weren't making investments."

Tony had to sell his prized party loft (at a significant loss) just to make payroll, but they made it round the bend. "Our customers didn't know that we were struggling with cash flow and so on. So, customers kept coming. Our revenues were going up, and so we knew that there was definitely a potential business here. So, it was never a question of whether the idea would work. It was just whether we could get over that cash flow hump."

In some ways the trends happening today in social networking seem like they were designed for the transparent CEO. Twitter, for example, was a no-brainer for Zappos. But other trends, like virtual offices, don't work for Tony. He says a community needs proximity, and for him Vegas was the right place to locate and build Zappos. Zappos isn't virtual, it's physical. "We really wanted to build the company around culture, company culture being the number one priority. And it's much easier to build a culture when it's actually in person versus remotely by email."

And while Zappos remains the company that will pay you to leave, Tony reveals that they haven't had to write a check like that in almost a year. Zappos, it seems, is a place folks aren't in a hurry leave once they get in the door. Tony says it's about Happiness.

"So many people when they go to the office, they leave a little bit of themselves at home, or a lot of themselves at home. And they have to put on this different persona in the office, especially in corporate environments. And our whole...there's a lot of talk about work life separation or balance and so on, whereas our whole thing is about work life integration. Its just life."

As you read Delivering Happiness, it's clear that Hsieh is talking about customer happiness, but also employee happiness, and even his happiness. He says the goals of Happiness aren't mutually exclusive.

"There's a lot of talk about work life separation or balance and so on, our whole thing is about work life integration. Its just life. And so the ideal would be if you can be the same person at home as you are in the office, and vice versa. And when people actually feel comfortable being themselves, so much creativity comes out of that."

So, if you believe Hsieh, there really is a change in the wind. A change in the way companies think, and act, and those that understand that will survive and thrive.

"I think we're just at the beginning where companies are becoming more and more transparent whether they like it or not. People are becoming...just because the information is everywhere and it's pretty hard to control now. So I think moving forward it's going to be only the authentic companies or people can win because everyone else will eventually be outed."

Can it be that Happiness is profitable?

The book is titled Delivering Happiness and the subtitle is A Path to Profits, Passion and Purpose. Tony Hsieh says in the book that research found that the best companies in terms of long-term financial performance are ones that are able to combine profits, passion and purpose.
If that's true, then we're standing at the edge of a new Happiness Culture, and it all began with selling shoes online.

"There's three types of happiness and really happiness is about being able to combine pleasure, passion, and purpose in one's personal life. I think it's helpful and useful to actually think about all three in terms of how you can make customers happier, employees happier, and ultimately, investors happier."

Watch all the Tony Hsieh interview videos on Curation Nation. Follow author Steve Rosenbaum @Magnify

Lecture 10: Don't Fuck Up the Culture

https://medium.com/@bchesky/dont-fuck-up-the-culture-597cde9ee9d4

Brian Chesky

On Monday, October 21, 2013, I sent this letter to our entire team at Airbnb. I have decided to publish this in the event it is helpful to entrepreneurs building their cultures.

Hey team,

Our next team meeting is dedicated to Core Values, which are essential to building our culture. It occurred to me that before this meeting, I should write you a short letter on why culture is so important to Joe, Nate, and me.

After we closed our Series C with Peter Thiel in 2012, we invited him to our office. This was late last year, and we were in the Berlin room showing him various metrics. Midway through the conversation, I asked him what was the single most important piece of advice he had for us. He replied, “Don't fuck up the culture.”

This wasn't what we were expecting from someone who just gave us $150M. I asked him to elaborate on this. He said one of the reasons he invested in us was our culture. But he had a somewhat cynical view that it was practically inevitable once a company gets to a certain size to “fuck it up.” Hmm.. How depressing I thought.

Were we destined to eventually “fuck up our culture?” We talked about it a bit more, and it became clear that it was possible to defend, and actually build the culture. But it had to be one of the things we were most focused on. I thought to myself, how many company CEOs are focused on culture above all else? Is it the metric they measure closest? Is it what they spend most of their hours on each week?

Culture is simply a shared way of doing something with passion.

Our culture is the foundation for our company. We may not be remembered for much after we are gone, and if Airbnb is around 100 years from now, surely we won't be a booking website for homes. We will be far past this in our evolution (not to mention that kids 100 years from now will be asking their grandparents what websites were).

The thing that will endure for 100 years, the way it has for most 100 year companies, is the culture. The culture is what creates the foundation for all future innovation. If you break the culture, you break the machine that creates your products.

So how do we build culture?
By upholding our core values in everything we do. Culture is a thousand things, a thousand times. It's living the core values when you hire; when you write an email; when you are working on a project; when you are walking in the hall. We have the power, by living the values, to build the culture. We also have the power, by breaking the values, to fuck up the culture. Each one of us has this opportunity, this burden.

Why is culture so important to a business? Here is a simple way to frame it. The stronger the culture, the less corporate process a company needs. When the culture is strong, you can trust everyone to do the right thing. People can be independent and autonomous. They can be entrepreneurial. And if we have a company that is entrepreneurial in spirit, we will be able to take our next "(wo)man on the moon" leap. Ever notice how families or tribes don't require much process? That is because there is such a strong trust and culture that it supersedes any process. In organizations (or even in a society) where culture is weak, you need an abundance of heavy, precise rules and processes.

There are days when it's easy to feel the pressure of our own growth expectations. Other days when we need to ship product. Others still where we are dealing with the latest government relations issue. It's easy to get consumed by these. And they are all very important. But compared to culture, they are relatively short-term. These problems will come and go. But culture is forever.

Brian
11. Company culture & building a team II
Lecture 11: Company Culture and Building a Team, Part II

http://startupclass.samaltman.com/courses/lec11/

Ben Silbermann, John and Patrick Collison

Sam Altman: Part two of Culture and Team. We have Ben Silbermann, the founder of Pinterest, and John and Patrick Collison, the founders of Stripe. Founders that have obviously, some of the best thinking about culture and building their teams.

There's three areas we are going to cover today. One will be general thoughts on culture as a follow up on the last lecture. And then we are really going to dig in to the findings of these companies and building out the early team. Then how that changes and evolves as these guys have scaled their companies to the hundred plus, I don't even know how many employees you have now but quite a lot, these very large organizations and how they adapt these principles of culture. But to start off, I just want to ask a very open ended question which is, what are the core pieces of culture that you found to be the most important in building out your companies?

**Ben Silbermann:** Sure. What are the most important parts? For us, we think we think on a few dimensions. One is who we hire, what those people value. Two is what we do every day. Why do we do it? Three is what we choose to communicate and I think four is how we choose to celebrate. Then the converse of this is what you choose to punish, but in general I think running a company based on what we celebrate is more exciting than what we punish. I think, the four things I think make up the bulk of it for us.

**John Collison:** One thing, I think Stripe has placed a large emphasis on, more so than other companies, is transparency internally. I think it's something been really valuable for Stripe, and a little bit misunderstood. All the things people talk about, like hiring really great people or giving them a huge amount of leverage.

Transparency for us plays into that. We think that if you are aligned on a high-level about what Stripe is doing, if everyone really believes in the mission, that if everyone has really good access to information, and everyone has a good picture of the current state of Stripe, then that gets you a huge amount on the way there in terms of working productively together. And it forgives a lot of the other things that tend to break as you grow a startup. So as we have grown, we started off two people, we're now a hundred seventy people, we've put a lot of thought into the tooling that goes around, transparency. Because with a hundred and seventy people, there's so much information being produced that you can't just consume it all as a fire hose. And so, how we use email, things like that, we can go into it more later. But that's one of the core things that helped us work well.

**Patrick Collison:** I think culture, to some degree is a resolution to a bandwidth problem. In the sense that, when you start out working on something you are coding all the time, but you can't code all the things that you think the product may need. And so the organization gets larger. Maybe in some idealized world, I don't actually think this is true, but ideally you should be involved in every single decision, in every single type of moment of the company, everything that happens, but obviously you can't. Or maybe you can if you're two people, but you certainly can't if you're five or ten. At that time, it comes very quickly, by a hundred and fifty, it's completely hopeless.

And so culture is the invariants that you want to maintain, as you get specifically involved in fewer and fewer decisions over time. When you think about it that way, maybe intended importance becomes self-
evident. Again, the fraction of things you can be involved in directly, diminishing, exponentially, assuming your headcount growth is on a curve that looks like one of the great companies. And yeah, that's super important. It manifests itself in a bunch of different ways. For example, in hiring, maybe the reason the first ten people you hire, the decisions are so important that aren't just hiring those first ten people, you are actually hiring a hundred people because you think each one of those people are going to bring along another ten people with them. And thinking exactly what ninety people that you would like those first ten people to bring on. It's going to be quite consequential for your company but really briefly I think it's about abstraction.

Sam Altman: One thing a lot of speeches in this class have touched on is hiring those first ten employees. If you don't get that right, the company basically never recovers. But no one has talked about how to do that, so what have the three of you looked for when you have hired these initial employees to get the culture of the company right? How have you found them, and what have you looked for?

Ben Silbermann: I guess this answer is different for every company. I'll say for us it was very inductive. Clearly I looked for people that I wanted to work with, and I thought were talented. I have read all these books on culture, because if I don't know something, the first thing I do is go read about it. Everyone has all these frameworks, so I think one big misconception that someone said once was, people think of culture as architecture when it is a lot more like gardening. You plant some seeds, then you pull out weeds when it's not working, and they sort of expand. When we first hired people, we hired people that were more like ourselves. I often looked for three to four things that I really valued in people. I looked for people who worked hard, had high integrity, low ego. I looked for people who were creative, super curious, which meant they had all these interests.

Some of our first employees are some of the quirkiest you have ever met. They were engineers but they had all these crazy hobbies. Like one guy made his own board game, with his own elaborate set of rules. Another guy was really into magic tricks, and he not only coded a magic trick on an iPhone but he shot the production video in the preview. And I think that quirkiness is a calling card that we find, the people that are excited about many disciplines and extraordinary at once, tend to build really great products and are really great at collaborating.

And the last thing, we really want someone who wants to build something great. And they aren't arrogant about it, they want to take a risk and build something bigger than themselves. And that, in the beginning, is very easy to select for. If you were in our position, we were in this horrible office, nobody got paid. There was no external reason to stay except wanting to build something to join. In fact there was every reason not to. And that's something, looking back, that I really value. Because you always knew people were joining for the purest of reasons and in fact forgoing other job opportunities, market salary, a clean office, good equipment just for the chance to work here. To this day, I think a lot of those traits are seeded and embedded in the folks that we look at now.

John Collison: The first ten hires is really hard, because you're making these first ten hires at a point where no one's heard of this company before. Nobody wants to work with you. You are these two weird people working on this weird idea.

Patrick Collison: Their friends are telling them not to join. For our second employee, he either accepted the offer or was just about to, and his best friends took him out the night before, it was a full on assault, why you should not join this company. Why it is ruining your life basically. And the guy continued to join, actually one of those friends now works at Stripe, but this is what you are up against.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
**John Collison:** And it's hard, no batch of ten people will have as big of an influence on the company as those first ten people. And I think that everyone's impression of recruiting is you open LinkedIn and it's sort of like ordering off the dollar menu, I want that one, that one, that one and now you have some hires. At least for us, it was over a very long time period talking to people we knew, or friends of friends into joining. We didn't have huge networks, we were both still in college by then. So there were really no people that we worked with to draw in. So a lot of those early Stripes were people we heard of from friends.

The other interesting thing they had in common, they were all really early in their career or undervalued in some way. Think about it, if someone is a known spectacular quantity, then they are probably working in a job and very happy about it. So we had to try to find people who were, in the case of our designer that we hired, he was eighteen and in high school and in Sweden at the time. As the case with our CTO, he was in college at the time. A lot of these people, they were early on in their careers, and the only way we could, you can relax one constraint, you can relax the fact that they are talented, or relax the fact that it's apparent that they are talented. And we, not consciously, we relaxed the latter.

**Patrick Collison:** Finding the right people, you have to think like a value investor right, you're looking for the human capital that's significantly valued by the market. You probably shouldn't look to hire your friends from Facebook, and Google, or whatever, they are already discovered, and if they want to join you, that's great. They are probably harder to convince. John has spent a little while yesterday afternoon trying to figure out in retrospect, what traits out first ten people had in common and felt were significant. Generally speaking in culture I want to carry out everything we say, advice is very little extrapolated and I think there is a lot of truth to that.

For our first ten people, the things that seemed to be important, they were also very genuine and straight. And I think that matters quite a lot, there are people that others want to work with. That there are others that people trust, that they are intellectually honest on how they approach problems. They are generally people who like to get things finished. There are a lot of people who are really excited about tons of things. Only some of those are excited about completing things. There is a lot of talk out there, like hiring people off their GitHub resumes, that doesn't really ring correct to me as there is a large premium on lots of different things. I think it's much more a priori, much more interesting to work with someone who took two years to spend time going deeper into an area. And then the third trait that we looked for is that they cared a great deal, it's offensive to them when something is just a little off.

In hindsight there were all these crazy things that we use to do that, do in fact, seem crazy. Like I should have not done them. Everyone was always like, it was borderline insane how much they cared about tiny details like we used to. Every single API request that ever generated an error went to all of our inboxes and phoned all of us. Because it seemed terrible to get an error that didn't get a resolution from the users standpoint. Or we used to copy everyone else on outgoing email and point out slight grammar or spelling mistakes to each other. Because it would be horrible to ever send an email with a spelling mistake. Anyways, those are the three traits we came up with, genuine, caring a great deal, and completing things.

**Ben Silbermann:** I have something to say, I don't think there is a wrong place to find people. So when I look back at our first hires we hired, they came from all over the place. I put up ads on craigslist, I went to random Techtalks, we used to throw weekly BBQ's at the office, bring your own food and drinks and then we would just talk to folks. I think every time I went to get coffee at Philz, one of you guys were recruiting. Because your office was strategically placed next to the best coffee shop. But I think
the really good people, generally are doing something else so you have to go seek them out instead of expecting that they are going to seek you out. Triple when no one has ever heard of or is using the product that you work on.

**John Collison:** Yeah and it's probably really important to have a great elevator pitch, not just for investors but because everyone that you run into right now is six months to a year down the road a recruit. So the right time to have gotten them excited about your product, the right time for them to have started following us, is as soon as it can start. It's going to take a very long time to recruit people, so getting people consistently excited about what you are doing will pay back later.

**Patrick Collison:** Maybe this is a little tangential, but a bunch of our friends started companies right out of school. And we started thinking, what goes wrong in those companies? And I think the most common failure mode was doing something overly niche, overly specific or bad. I think there is a major shift in time horizon as you go from classes to a startup. A class plays out on a quarter, or a semester, where a startup is a five or ten year thing. And I think this is really problematic, because it's really quite hard to hire people for niche things. If you told someone, look we are going to build a rocket that goes Mars, that sounds almost impossible but sounds fucking awesome. It's really easy to convince people to work on it. Instead of, well we are going to work on, I'm not going to give any specific idea, probably going to sound like we are doing a startup. But if you pick something pretty narrow, generally comes out of this class project that's actually much harder to hire for.

**Sam Altman:** One specific question that has come up a lot is how, as a relatively inexperienced founder, you identify who the really good people are. So you meet people at these BBQ's, you are friends, maybe you have worked with them. What did you guys do specifically to identify that this person was going to be really great? Or did you really get it wrong? When did you learn you could identify raw talent? Or say they work at Google, or Facebook so they must be good.

**Ben Silbermann:** You will never 100% know until you work with folks. So the flip side is, if the person you hired is not a good fit, you owe it to them and to the company to tell them where to improve and if they aren't working out, then to fire them. But I think, generally, the question of talent falls into two big buckets. One is, you have some sense of what makes them good at their job. And there are some areas where you can test that area. And there are some that you don't. And the ones where you don't are much more difficult. So what we would do is a few things.

Before we talk to anyone, we try to figure out what exactly is world class in that discipline need. And this comes in a little later when you are hiring the head of finance and you know nothing about finance, except what is contained in a library book that you got. Like an introduction to finance or marketing. So I always made it a habit of mine to talk to people I knew de facto were world class and just asking them, what are the traits you look for? What are the questions you ask? And how to find them? If you are looking for the next person that is as good as you, where is that person working right now and what's her phone number? I think that learning what's good and bad during the interview process is extremely expensive. It is an expensive use of your time, expensive use of everyone else's time. A recalibration of that really matters.

Then once you have someone in an interview process, you will build the process over time to screen quality. Pinterest, we have an evolving set of questions that are rotating through and we are always asking if these are good indicators or bad indicators of quality. The other thing the questions are supposed to do is give us a sense of, is this the right place for this person to come in and work? This is the point you guys made about being very transparent. About what's going to be easy or hard. Really great people want to do things that are going to be hard. They want to solve tough problems, so there
was a sense of brilliance in Google sending out these interview questions that were thought to be really difficult. Then people who like solving problems, they come out and seek those. I think it's really important as companies get bigger that they don't whitewash the risks. I heard that Paypal, you go in and after the interview they say, by the way, Mastercard wants to kill us and you will be doing something that is illegal, but if you succeed you will redefine payments. Or when they were recruiting for iPhone, they didn't even tell people what they were doing. You won't see your families for three years, but when you are done, your kids, your kids' kids will remember what you built. I think that's a really good thing in recruiting as well. Be very very transparent on why you think it's a great idea, but you lay out in gory detail why it's going to be hard. And then the right people select in or they select out of that opportunity.

Patrick Collison: Evidence suggests they were able to see their kids though.

John Collison: I think one thing you have to do when identifying talent is have the confidence to interview in a way that works for you. I think, say you are not the world's best engineer and you are trying to interview engineering candidates. I thinks it's tempting to co-opt what everyone else does, get them to put things on a whiteboard and do other engineering things. In the case of Stripe, we flew a guy out and we spent a weekend coding with him and looking over his shoulder. It was the only way we could tell and get ourselves confident if that guy was any good. And I think you can extend that to any roles you are not an expert in. In that I am no business development guru but when we hire for business development roles, we have a project that we have them talk about, how they would improve an existing project that Stripe has or which new projects they would go out and do. And even if it is not my domain area, I am confident enough I can judge those really well. I think people often have this imposter syndrome when it comes to interviewing for roles.

Patrick Collison: I think a specific tactical thing to do, again, for the first ten people is to work with them as much as you can before committing to hire them. Once you hit a certain scale it's kind of impractical to put them on that side and be unskilled. Expensive from your side. But it's really worth it to the first ten people, right. In the majority, the first ten people, we worked with them to some capacity for a week in advance. It's pretty hard to fake it for a week, it tends to be quite clear quickly. Another answer I thought of to the question, how do you know if someone if great? And people talk about this notion of the 10x person or what the skill set is. I don't know what 10x means. I think the slightly more intuitive decision is, is this person the best out of all of their friends at what they do? It's a little insensitive on how they choose among friends, but for me at least I find that a better way to think about it is, is this the best engineer this engineer knows? And the other thing worth mentioning is, on the first ten people on the culture and team topic, I think everyone doesn't realize until they go through it themselves, how important it is because in life and media people focus too much on founders. Here we are and we are reinforcing the structural narrative that Stripe is about John and Patrick and Pinterest is about Ben. When the vast majority of what our companies do, 99% are done by people that are not us, right? It's obvious when you say it but it's very much not the macro narrative. These are abstracts and you associate them with certain people. For companies like Apple and others, Steve Jobs was a tiny, tiny part at the end.

John Collison: So don't screw up is what you are saying?

Patrick Collison: Something like that.

Ben Silbermann: I think referencing people is really important. Referencing people is just what it sounds like. Asking people with experience for their honest opinion. We do that really aggressively but we are trying to figure out what this person is like to work with. We are trying not to validate if they
told the truth on their resumes because we assume they told the truth. So a very standard question is in an interview I might say, hey we both know Jonathan, because we are both friends, if I asked him what you are best at, what you are the most proud of, or what you were working to improve, what would he or she say? Because it creates social awareness and accountability. Then I typically ask something that makes the question, that is typically soft, feel a little bit more quantitative and then calibrate that over time. To evaluate this person's dimensions, is this person the top 1% of the people you worked with, the top 5%, and the top 10%? And it forces scarcity that gives them material reference. Instead of saying, hey what's the best thing about John? You say, he told me he was good at these things. Can you validate it? Yeah, sure. It's the type of tool that should be taken seriously.

John Collison: And referencing, obviously, isn't easy to begin with. But it does provide really useful over time. And I think for name references people do want to be nice so you have to create an artificial scarcity by saying, where would you rank this person with people you worked with. You should aim to spend fifteen minutes on the phone with that person instead of letting them say, yeah this person is awesome.

Sam Altman: Also those references are a tremendous source of recruits. Once you hire these first people and they join, what have you done to make them effective quickly and to get them to the right culture place? Hiring is usually difficult but then not as difficult as making them happy and effective. So what do you do with these early employees to accomplish that?

Ben Silbermann: Well the answer to that has changed since we went from small to bigger. When we first started it was because we needed that person, a long time ago. So their whole onboarding was, here's your computer, we already set up your environment, don't worry about it, this is the problem we have to solve together. That's the nature of the startups, we were all in this tiny two bedroom apartment. All the other things, building personal relationships, spending time together, it all happened magically. You didn't have to do anything.

The one thing I would like to add to that is we always reminded people where we want to go with that someday. Because it's really easy to drop someone into a problem and they would think the whole world was this little problem in front of them. We would always say, someday we want to do for Google what they did for search. Our plan for trying to get that done.

Now as the company grows, I think that problem has to get a little more formalized. So we spend a lot of time thinking and constantly trying to refine what that person looked like from the day they came in, to their first interview, through 30 days after they joined. Do they have someone's name they know? Do they know who their manager is? Have they sat down people on their team? Do they know what the general arch of the company is? And what the top priorities are? And we have a program we do. It's a week long and they have functions to go in deeper. And that's something that has always been refined. And the output metrics on that is one, we ask people, what did you think afterwards, then 30 days afterwards? Then we also ask their peers and manager, hey is this person up to speed? Do you feel we did a good job at making them productive? If we haven't then thats a key that a) we should not be hiring any more people because we're not doing a good job bringing in new people and b) we need to retool that.

I think those things are important. I just wouldn't discount how important it is to get to know the person as a person. What's their aspirations? What's their working style? How do they like to be recognized? Do they really prefer being in total silence? Are they a morning person or night person? Knowing those things, it just demonstrates that you care about them individually and collectively, what your goals are.
John Collison: I think there are two things that are important at any stage, though the implantation will change. First is to get them up and running quickly to do the work. That is how you are going to find the problems, it is how progress is measured in the real work they are doing. And so when we have engineers start, we try to get them committing on the first day. When we have people in business roles start, we will have them in real meetings the first day on what they are meant to be working on. Sometimes it's easy to be tentative and ease people in. We are much more, push people off the cliff. Then second, we try to quickly give people feedback. Expectedly giving people feedback on how to adapt to the culture. When you think about it, if you have built a strong culture as all the companies up here are trying to, it's going to take some adapting from the person, it's not going to be necessarily easy. One thing we have at Stripe is the culture is a lot more written. So you have people next to each other, with headphones on, IMing each other. And for a lot of people coming in and working in an environment like that it's sort of hard-

Patrick Collison: In from normal places.

John Collison: Exactly yeah. So from everything high level of how you are doing at your job to minor cultural issues, the more feedback you give them, the better they will do. Its unnatural to be telling people if they are doing a good or bad job. You don't do that in your normal life, hopefully you are restrained. But when you have employees that is what you owe them to do well.

Sam Altman: So I think this is a good transition to when your companies have scaled. What are the biggest changes you have had to make to your hiring policies and to how you manage the teams as you have gone from two to ten to a thousand employees?

Ben Silbermann: There are a lot of changes. I think one thing we try to do on the team side is to make the teams feel as autonomous and nimble as possible within the constraints of the organization. That means over time we are trying to make it feel like a startup of many startups. Rather than this model setup with form policies cut horizontally through it. It's easier said than done, I don't think we are all the way there but one goal is that each team has control, to hold the resources that they need to get the goal done. They know what the most important thing is and how it's measured. That way the management problem because somewhat tractable. Otherwise it feels completely impossible if you can't decompose it into atomic components. You just look at it like, Oh my gosh, complexity level is rising geometrically, on a management level, it's never going to work.

You have to create these abstracted units. At least that's what we are going to try to do. Pinterest in particular, the real challenge with building those abstract units is, we want units that encompass a super strong designer, or a super strong lead engineering, a writer, often times a community leader. We want them to be self contained. That kind of makes it hard, but that kind of core is to our philosophy to build products. We put people together that have all these kind of disciplines, lots of things then we anchor them to a certain project then we try to remove barriers to let them go fast. Then we find no barriers and we sit down to figure out how we speed that up. I think hiring is a little different. I think the biggest change and the biggest asset you get is the people, referrals become more and more the life blood depending on the network of the people you bring in. So one of the lucky and in hindsight decisions we made was our actual fourteenth or fifteenth person we hired was a professional recruiter. She worked at startups,, she worked at big companies like Apple. But she sort of knew where that pipeline breaks down. Knew the early indicators, and taught everyone not just how to screen for talent but to identify the people who are going to be culturally really good for the company. And I think looking back on that, its something I personally really value.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Patrick Collison: There is a huge amount of stuff here that is under management growth. Either your company fails really quickly or all of your problems become about management growth. One thing that tends to take people by surprise, and took me by surprise, is how quickly the time horizons change. In your first month, you are largely thinking about things one month ahead, right? Maybe that is what your development road map is oriented around. Who you are working with, maybe it's a very informal relationships where they haven't fully committed to be full time or not. The more time that goes by, I feel that has a reciprocal in the time horizons.

In one year, you are thinking a year ahead. After four years you are thinking four years ahead. That increases very quickly right? After one month it's super short term. After eleven months, you should actually be thinking and planning a year ahead and think about human structures. Thinking about stuff Ben talked about. Where you want to be going long term. Things like that. I think that plays into the hiring and that in the early days you have to hire people who are going to be productive. Essentially, you don't have the luxury of hiring people that look to be promising but they are not going to be up to speed for another year or two. They have to be able to work immediately. But after two or three years, then it becomes much more reasonable to make those investments.

Actually if you are not making those investments you are probably being much too short term. And so I think that's really important. All of these problems in some sense are easy. Like, how do you build such good bonds with people? We all do it every day. How do you make it systematic, and effective at scale? It's all going to vary significantly, in perfect proximation of what you would ideally do small, what hacks can you pull? Make it work as well as you possibly can at a large scale. A rapidly growing company, with head count here two-three heads a year, it's a very unnatural thing. What's the least bad way of managing that growth? I think it's worth being systematic about it, thinking of ways to do it. Realizing you can't do much better than ask a question. For Stripe, it's things like, we have three meals a day at long tables where everyone can sit together, if you think about how much more human interaction happens with having these randomly mixed meals. It's vast, right? A whole list of things like that. I think that's the general framework.

Ben Silbermann: One thing I'm really curious about. You guys value transparency. Have you scaled it over time? I know with us, we think about it all the time. Just curious.

Patrick Collison: Startups, I can't remember who defined it as, a startup is an organization that is not yet stuck with all these principal agent problems. That large companies, what is locally optimal for you frequently is not what is globally optimal for the company. As a consequence of that, a reason a startup can just work differently than a big company, at a big company, a lot of the things that are good for you, you couldn't do them in a completely transparent environment because people would think less of you. But because everyone is rolling in the same direction, a startup, you can kind of make all the information transparent. Like I said earlier, Stripe used to bcc us to be on every email unless you opted out of it. We thought that would be more efficient, you wouldn't need to have as many meetings if you could keep abreast of what is happening. And over time, we sort of built an interesting framework of mailing lists. We now have a program for generating gmail filters. For a rocky path of fifty people or so to Ben's point of asking people how they are getting along after several days. They all reported terrible because they couldn't find all the emails people were sending to them. They were missing things and everything.

John Collison: Gmail broke at one stage.

Patrick Collison: Right, right. At one point Gmail broke because we were sending too much email. It is hard to scale because you might contact somebody out of the company with some great idea. The
person sitting across the way from you thinks it's the stupidest thing they have ever heard. You are kind of under that scrutiny of the whole organization in some degree with all your communication. That's the challenging side, that people more formally know what's happening. I don't feel I can give a stronger endorsement of it than it has worked so far. I really am curious how it will work when we are five thousand people, if we are ever at that scale.

**John Collison:** I think a couple things that helped us scale it, is we changed tools in changing the culture around it. On the tools front it use to be the case you could keep abreast of what's happening in the company by reading all the email. Now we weekly all hands on deck. We actually have to put all this work into communicating what is going on in this company. Since there is so much more. The second is, cultural side so much information in terms. You have to create social norms around it. Obvious things such as what is confidential to Stripe. Less obvious things, like when emailing someone or talking in Slack or IRC that is now viewed by one hundred seventy people, it's pretty easy to get stage fright. And it's pretty easy with what you thought was a reasonable proposal, you get this drive by criticism and you are less likely to share in the future. We have had to create norms around when it is okay to jump into discussions and how that interaction works because people are around the stage much more.

**Patrick Collison:** I'm sure it's not good to put anyone on the spot, but Emily interned at Stripe this summer. I am curious, as an intern. what you thought of it.

**Emily:** Overall it's great. The first week, I spent most of my time trying to get caught up on what the company was doing. It can often be quite distracting from your own work as often times there are other parts of the company that you are interested in.

**Patrick Collison:** By the way. Google Docs is like news feed, you can see all the documents.

**Emily:** And you are encouraged to make everything public, everything that you work on. But overall it gets you sped up rather quickly. There are these things called Spin Ups, where every leader of the team gives a thirty minute talk on what they are doing and how you can contribute if you are interested.

**Patrick Collison:** Do you think transparency was that good?

Emily: Yes, I remember having a hard time remembering what I should and should not subscribe to. The first week having two thousand emails in my inbox, then by the end there are three or four teams you want that information coming up from.

**Sam Altman:** So this question is for Patrick and John. Have the people that you hired early been able to grow up into leadership roles?

**John Collison:** In Stripe's case yes. A lot of the first ten people are in leadership roles now. I think that's one thing that corporations, it's an unnatural skill that they need to get good at. People don't exactly come out of the womb being good at management or at leadership. And being able to develop that in people and helping people progress as they spend a number of years at the company, it's a lot of work when people are running around with their hair on fire. But it's also damaging if the company can't develop that skill.

**Ben Silbermann:** I think for us the answer is some yes and some no. I think one of the benefits of working at a startup is you can be handed a challenge no one else would be crazy enough to let you take on. And that could be managing people, it could be taking on a project. And also if you ask someone to take a risk like that, it shouldn't be one way through the door if you don't succeed.
Otherwise it creates fear to give it a shot. So we have some folks managing a large team at the start, individual programmers, individual engineers. And they say, Hey I would love to try leading a project, leading a group, then taking responsibility for management. Then we have other folks that try it and are really glad they did so they know they never want to do that again. We try that, for those people, you can have just as much impact on the company through your individual contributions as an engineer or what have you. But it's really hard to predict unless you give people a shot. So my strong preference is you give as many people a shot as possible. And in the few areas where you feel there is too much of a learning curve relative to the business development you are trying to achieve, that is when you look for someone who might walk in and really execute well on the job.

So the question was, has the vision changed since we initially started? Well I think on the vision, when we first started hiring, we were like we are going to build this really cool tool, people are going to enjoy it. I like collecting things, maybe others like collecting things. And what we didn't expect that revealed itself early on, was that looking at other peoples collections was, it turned out to be this really amazing way of discovering things you didn't know you were looking for. It becomes a solution to the problem that a lot of other technologies don't have.

Over the last year we have poured a lot of resources into building our recommendation products, search products, feed products. Leveraging the unique data that we have, which are these pins that were all picked by someone and hand categorized. And the on the audience side, I think the first big surprise was truthfully when we first started, we didn't know if anyone would really use it. eE were just happy that anybody who wasn't related to us or obligated would use it. The biggest surprise has been the diversity of people and how diverse those groups have been. And I think that's been one of the things that's the most exciting. And the funny thing is, often as the company goes further along your aspirations get bigger. There is this gap that exists, and I tell my team, between where we are and we should be. Objectively we are further along, I feel the gap has widened. But I think that's a really common trait among people who found companies.

Sam Altman: So the question is, most start ups are not the iPhone. You can't guarantee that most people's grandchildren are going to remember this because most startups fail. How do you convince people to make sacrifices to do join a startup?

Patrick Collison: I think part of why it resonates with people is because it's not guaranteed. If it was it would be boring. There is the prospect of affecting this outcome, but nothing more than that potential. As far as not seeing their families and kids, startups do involved longer hours in the beginning but I think that story is overstated. Even the startups that in the earlier days had some sort of longer working days, have a tendency to exaggerate. It's kind of like the startup version of fishing. Every startup thinks they worked more insane hours than the next one back in the early days. It's like we literally never slept for two years. I think realistically for most people, it's not that big of a sacrifice. I think on average, people work on average two hours more a day. It is a sacrifice but it is not forgoing all pleasure and enjoyment for the next half decade.

Ben Silbermann: Even the iPhone wasn't the iPhone before it got done. No smart person you are hiring thinks you have a crystal ball into the future that only you have and that joining is a guaranteed thing. And in fact if you are telling them that and they select in, maybe you shouldn't be hiring them because they didn't pass a basic intelligence test of certainty and the future. But I think it's fair to say what's exciting and where you think you can go. And where it's going to be hard and chart your best plan. And then tell them why their role in it can be instrumental because it is.
I really liked what you said, if you tell people, Hey we are going to go to Mars, it attracts the best people and you are closer to Mars and they know that. What I would discourage is whitewashing all that. And if people are joining because they want all the certainty of Google and the perk of working in a small startup with more email transparency, then that's a really negative sign. For example, when I interview people, they often say, I'm really passionate about what you are doing. I often ask where else they are interviewing. If they list seven companies that have nothing to do with each other, except they are at the same stage, I love the stage of discovery, so I'm interviewing at Stripe, Jawbone, Airbnb, Uber, I'm also putting my resume into Google X, that's a sign they are probably not being authentic, which you care about. And those folks, when things get hard, they won't stick it out and work through it, because they were really signing up for an experience, not for achieving a goal.

Patrick Collison: I think the other thing that motivates people a great deal is the prospect of affecting some outcome, is just the personal development angle. And a startup because it's more lightly staffed it's much less forgiving. Whether or not you are the best or the worst person in the world, you are probably not going to alter Google's trajectory. Whereas if you really want to benchmark yourself and see how much of a contribution and impact you can make, the startup is a much better place to test that.

Sam Altman: How does your user base affect your hiring strategy?

Ben Silbermann: Conventional base, you only hire people who use your product religiously everyday. And that probably works well if you make an API. For us, we screen for people who have vision and discovery online. And they have to know how our service works, and they have to have used it. But they may not be a lifelong user. And for us that's great, we can ask what is the barrier that is preventing you from using it? Come join, we will move that barrier. Help us get closer to that vision. If you read a startup book, there is all this wisdom, but it is only useful if it works in your certain circumstance. So for us, we have had to broaden the lens a little bit and bring in people who are excited about the mission, that care about our product and our approach to building products. Even if from day one they weren't our earliest users.

John Collison: The one thing I want to tack onto this is, we touched on it being hard to hire early on for those first employees, you have people with other options, you are very much at the ugly duckling stage. Hiring people who are passionate about your product is a great way to find people. You have a natural advantage over other companies. I know in Stripe's case, we hired four Stripe users, people who we probably couldn't have gotten otherwise. I'm sure it was the same in Pinterest's case where you will get all this benefit at working with Pinterest, like, Hey it's Pinterest.

Sam Altman: Thank you guys very much for coming in today.
Lecture 11: What it's like to work for Stripe

http://blog.alexmacaw.com/stripes-culture

Alex Maccaw

A company’s culture is something intangible and nebulous, and yet it can be just as important to success as revenues or growth. Culture influences everything, from design and product implementation to the level of support and operations of a company. It’s crucial to get it right.

I’ve been at Stripe for a few months now, and I’ve wanted to write about what it's like to work there. I’ve never been more impressed by the mechanics and culture of a startup. In fact, I’ve never seen anything quite like it.

Culture can be hard to define, and it certainly can’t be created with mission statements and performance reviews. However, there are certain steps that founders can take to create an amazing place to work. With that in mind, I’d like to give you a taste of what it’s like to work at Stripe.

Email - complete transparency

By convention, every email at Stripe is CC-ed to lists that go to either the entire company or to any particular team. This includes internal person-to-person correspondence. Our lists include dev, sys, office, product and support. From the internal wiki: It turns out that Stripe generates a lot of email. In most cases, this is quite an intentional, positive thing — it’s a great communication mechanism, persists forever, and is easily searchable. The reasons for CCing lists, is it's a really low-friction way to keep everyone in the loop, and that way people can jump in with helpful advice.

It’s also a really good way to preserve openness as we grow (everyone still gets to see all the cool and important things that Stripe is up to) without requiring much extra effort.

This requires a lot of filtering, of course, but it allows me to dip in and out of the company’s fire hose whenever I want. It gives every one of us a tremendous perspective and insight into what other people are working on, and a feeling of connectivity to the rest of the company.

I’ve never before seen this level of access or trust at a company. Other companies preach fearless communication. Stripe practices it.

Group activities

Aside from mealtimes, Stripe has a social list, and there are often after work drink-ups, BBQs and parties. For example, we all went to a local theater recently for a showing of The Dark Knight.

In addition to that, once or twice a year we organize company-wide hackathons: we all head together somewhere far away and just hack on new Stripe-related projects.

Everybody does support

Every single engineer does support, on a bi-weekly rotation. Even the founders John and Patrick. We provide support over an IRC channel, email, and through Stripe answers.

I’ve found providing support to be one of the best ways to learn about the company, how everything works internally, and our customer’s needs. I’ve been writing a bunch of tools to automate some of the most common queries, like chargeback evidence, but at the end of the day we want to ensure that there’s always a human available.

Hackathons & Capture the Flag

Every so often, Stripe has Hackathons. On a weekend, we invite a few hundred or so people around, break out a truckload of Ikea fold-up chairs, open up the office, order sandwiches, and hack on whatever comes to mind.

With Capture The Flag, we host a series of challenges that involve exploiting security holes. In our first contest, we had people ssh into specially configured EC2 servers. In the CTF this week, we’re doing web vulnerabilities: CSRF, XSS, SQL injection and such.

Once a week, on every Tuesday, the company meets for an all hands. Each team explains what happened in the last week and what they’re planning for the next. Again, this is all about transparency and communication.

At the end of the meeting we address FUD, or fear, uncertainty and doubt. Anyone can raise objections and discuss their concerns. The idea behind this, is that it prevents problems from staying hidden in one small part of the company, and makes sure everyone’s thinking about how to solve the problems we have.

The is the only meeting that everyone is required to attend. Meetings are expensive, so during the week we try to limit them to as few as possible.
I should stress that these are all independent initiatives from people inside the company. Anyone can come up with ideas like these and implement them.

**IM & IRC**

Google Chat and group IRC rooms are as integral as email to internal communication. Instead of meetings, we just do short chat conversations. Rather than constantly interrupt people’s train of thought, we prefer to communicate asynchronously. Chat is often the happy medium between the latency of email, and the bother of in-person interruptions.

That said, sometimes it is much better to discuss things face to face. If someone hasn’t got their headphones on, then it’s totally acceptable to come over and ask them for five minutes of their time when they can spare it.

Also, an IRC bot messages me every six hours or so asking me to optionally reply with brief description of whatever I’m working on. It then displays what I wrote on some of our dashboards. This is rather useful, as it gives visibility at a company wide level into what everyone’s doing.

**Paper reading**

Once a week, a group of us take a technical paper and discuss it over lunch. This week’s was *Characterization and Measurement of TCP Traversal through NATs and Firewalls*, about TCP tunneling through firewalls. In previous weeks we’ve covered Bitcoins and C-Store. As someone who missed out on that side of university, I find these particularly interesting. Often the papers are from domains that I have absolutely no previous experience in.

**Induction**

New employees are asked to send in specifications for their “dream machine”, which is then built and waiting for them on their first day. It turns out that the average monitor size at a company is a good tell as to how much employers care about first-class equipment and, in turn, their employees. Stripe gets this right.

For their first few days, new hires spend each day working with a different engineering group: product, systems, growth, ops, support. This is a great way to get familiar with what everyone at the company does, learn the codebase, meet everyone else, and make sure they can move freely across the company if they want to do so later.

**People & teams**

None of this would matter if Stripe didn’t have fantastic people. Hiring well is the key to all of this, and people are the foundation of any company’s culture. Frankly, I’ve never seen a team like Stripe’s; we have the best people in the industry.

When you hire great people, you can afford to give them a lot of autonomy. Teams are made up of small numbers of people who propose ideas and then go off and implement them. There are no dedicated managers. So far, we have 31 employees. Each team is small and nimble, and there’s a heavy focus on shipping quality software.

Lastly, people are encouraged to be generalists, and not to be afraid of exploring unfamiliar domains. I’ve been doing a bunch more work with servers recently, and Darragh, who usually runs ops, has fixed the dishwasher.

**Patches welcome**

Creating a great culture is key to an enjoyable working environment and ultimately, I believe, a successful company. It’s a shame when that aspect gets blindsided in lieu of other parts of the business, an all too common occurrence.

We’ve learned a few things about cultivating culture at Stripe, and we’re constantly experimenting and evolving. I feel incredibly fortunate to work in such a place, and I’m hoping this piece will be useful for people who want to create a similar environment.

[http://startupclass.samaltman.com/](http://startupclass.samaltman.com/) (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Lecture 11: How to hire

http://blog.samaltman.com/how-to-hire

Sam Altman

After startups raise money, their next biggest problem becomes hiring. It turns out it’s both really hard and really important to hire good people; in fact, it’s probably the most important thing a founder does.

If you don’t hire very well, you will not be successful—companies are a product of the team the founders build. There is no way you can build an important company by yourself. It’s easy to delude yourself into thinking that you can manage a mediocre hire into doing good work.

Here is some advice about hiring:

* Spend more time on it.

The vast majority of founders don’t spend nearly enough time hiring. After you figure out your vision and get product-market fit, you should probably be spending between a third and a half of your time hiring. It sounds crazy, and there will always be a ton of other work, but it’s the highest-leverage thing you can do, and great companies always, always have great people.

You can’t outsource this—you need to be spending time identifying people, getting potential candidates to want to work at your company, and meeting every person that comes to interview. Keith Rabois believes the CEO/founders should interview every candidate until the company is at least 500 employees.

* In the beginning, get your hands dirty.

Speaking of spending time, you should spend the time to learn a role before you hire for it. If you don’t understand it, it’s very hard to get the right person. The classic example of this is a hacker-CEO deciding to hire a VP of Sales because he doesn’t want to get his hands dirty. This does not work. He needs to do it himself first and learn it in detail. Then after that, he should lean on his board or investors to give him opinions on his final few candidates.

* Look for smart, effective people.

There are always specifics of what you need in a particular role, but smart and effective have got to be table stakes. It’s amazing how often people are willing to forgo these requirements; predictably, those hires don’t work out in the early days of a startup (they may never work).

Fortunately, these are easy to look for.

Talk to the candidates about what they’ve done. Ask them about their most impressive projects and biggest wins. Specifically, ask them about how they spend their time during an average day, and what they got done in the last month. Go deep in a specific area and ask about what the candidate actually did—it’s easy to take credit for a successful project. Ask them how they would solve a problem you are having related to the role they are interviewing for.

That, combined with the right questions when you check references, will usually give you a good feel about effectiveness. And usually you can gauge intelligence by the end of an hour-long conversation. If you don’t learn anything in the interview, that’s bad. If you are bored in the interview, that’s really bad. A good interview should feel like a conversation, not questions and responses.

Remember that in a startup, anyone you hire is likely to be doing a new job in three to six months. Smart and effective people are adaptable.

* Have people audition for roles instead of interviewing for them.

This is the most important tactical piece of advice I have. It is difficult to know what it’s like working with someone after a few interviews; it is quite easy to know what it’s like after working with them.

Whenever possible (and it’s almost always possible), have someone do a day or two of work with you before you hire her; you can do this at night or on the weekends. If you’re interviewing a developer, have her write code for a real but non-critical project. For a PR person, have her write a press release and identify reporters to pitch it to. Just have the person sign a contractor agreement and pay them for this work like a normal contractor.

You’ll get a much, much better sense of what it’s like working with this person and how good she is at the role than you can ever get in just an interview. And she’ll get a feel for what working at your company is like.

* Focus on the right ways to source candidates.

Basically, this boils down to “use your personal networks more”. By at least a 10x margin, the best candidate sources I’ve ever seen are friends and friends of friends. Even if you don’t think you can get these people, go after the best ones relentlessly. If it works out 5% of the time, it’s still well worth it.

All the best startups I know manage to hire like this for much longer than one would think possible. Most bad startups make excuses for not doing this.

When you hire someone, as soon as you’re sure she’s a star you should sit her down and wring out of her the names of
everyone that you should try to hire. You may have to work pretty hard at this.

Often, to get great people, you have to poach. They’re never looking for jobs, so don’t limit your recruiting to people that are looking for jobs. A difficult question is what you should do about poaching from acquaintances—I don’t have a great answer for this. A friend says, “Poaching is the titty twister of Silicon Valley relationships”.

Technical recruiters are pretty bad. The job boards are generally worse. Conferences can be good. Hosting interesting tech talks can be good for technical hiring. University recruiting works well once you’re reasonably established.

Don’t limit your search to candidates in your area. This is especially true if you’re in the bay area; lots of people want to move here.

View candidate sourcing as a long-term investment—you may spend time now with someone that you don’t even talk to about a job for a year or more.

Use you investors and their networks to find candidates. In your investor update emails, let them know what kind of people you need to hire.

As a side note, if I were going to jump into the mosh pit of people starting recruiting startups, I would try to make it look as much like personal network hiring as possible, since that’s what seems to work. I’d love a service that would let me see how everyone in my company was connected to a candidate, and be able to search personal networks of people in the company (LinkedIn is probably good at this for hiring sales people but not very good at this for developers).

*Have a mission, and don’t be surprised at how much selling you’ll have to do.*

You need a mission in order to hire well. In addition to wanting to work with a great team, candidates need to believe in your mission—i.e., why is this job more important than any of the others they could take? Having a mission that gets people excited is probably the best thing you can do to get a great team on board before you have runaway traction.

As a founder, you’ll assume that everyone will be as excited about your company as you are. In reality, no one will. You need to spend a lot of time getting candidates excited about your mission.

If you have a good mission and you’re good at selling it, you’ll be able to get slightly overqualified people—although, in a fast-growing startup, they’ll end up in a role that they feel not quite ready for quickly anyway.

You should use your board and your investors to help you close candidates.

Once you decide you want someone, switch into closing mode. The person that the new hire will report to (and ideally also the CEO) should be doing everything possible to close the candidate, and talking to her about once a day.

*Hire people you like.*

At Stripe, I believe they call this the Sunday test—would you be likely to come into the office on a Sunday because you want to hang out with this person? Liking the people you work with is pretty important to the right kind of company culture. Only a few times have I ever seen a scenario where I didn’t like an otherwise very good candidate. I only made the hire once, and it was a mistake.

That being said, remember you want at least some diversity of thought. There are some attributes where you want uniformity—integrity, intelligence, etc.—and there are some where you want coverage of the entire range.

*Have a set of cultural values you hire for.*

Spend a lot of time figuring out what you want your cultural values to be (there are some good examples on the Internet). Make sure the whole company knows what they are and buys into them. Anyone you hire should be a cultural fit.

Andrew Mason says “Values are a decision making framework that empower individuals to make the decision that you, the founder, would make, in situations where there are conflicting interests (e.g. growth vs. customer satisfaction)".

Treat your values as articles of faith. Screen candidates for these values and be willing to let an otherwise good candidate go if he is not a cultural fit. Diversity of opinions and certain characteristics (e.g. you want nerds and athletes both on the team) is good; diversity of values in a startup is bad.

There are some people that are so set in their ways they will never get behind your values; you will probably end up firing them. As a side note, avoid remote employees in the early days. As a culture is still gelling, it’s important to have everyone in the same building.

*Don’t compromise.*

In the grind of a startup, you’ll always need someone yesterday and it’s easy to hire someone that is not quite smart enough or a good enough culture fit because you really need a specific job done. Especially in the early days, never compromise. A single bad hire left unfixed for long
can kill a company. It’s better to lose a deal or be late on a product or whatever than to hire someone mediocre.

Great people attract other great people; as soon as you get a mediocre person in the building, this entire phenomenon can unwind.

*Be generous with compensation packages, but mostly with equity.

You should be very frugal with nearly everything in a startup. Compensation for great people is an exception. Where you want to be generous, though, is with equity. Ideally, you end up paying people slightly below to roughly market salaries but with a very generous equity package. “Experienced” people often have higher personal burn rates and sometimes you’ll need to pay them more, but remember that great companies are not usually created by experienced people (with the exception of a few roles where it really matters a lot.)

I am sure I will get flamed for saying this, but it’s the right strategy—if you want an above-market salary, go work at a big company with no equity upside.

Ideally, you want to pay people just enough they don’t stress about cash flow. Equity is harder, but a good rule for the first 20 hires seems to be about double what your investors suggest. For a company on a good but not absolute breakout trajectory, some rough numbers I’ve seen are about 1.5% for the first engineer and about 0.25% for the twentieth. But the variance is huge.

Incidentally, a very successful YC company has a flat salary for effectively all of their engineers, and it seems to work well. It’s lower than what these people could get elsewhere, but clearly they enjoy the work and believe the stock is going to be worth a lot. The sorts of people that will take this deal are the kind of people you want in a startup. And unless something goes really wrong, at this point, these engineers are going to make way more money than that would have taken higher salaries elsewhere—not to mention how much better their work environment has been.

You will likely have to negotiate a little bit. Learn how to do this. In general, materially breaking your compensation structure to get someone is a bad idea—word gets out and everyone will be upset.

*Watch out for red flags and trust your gut.

There are a few things in the interviewing/negotiating process that you should watch out for because they usually mean that person will not be successful in a startup. A focus on title is an example; a focus on things like “how many reports will be in my organization” is an even worse example. You’ll develop a feel for these sorts of issues very quickly; don’t brush them off.

If you have a difficult-to-articulate desire to pass, pass.

*Always be recruiting.

Unfortunately, recruiting usually doesn’t work as a transactional activity. You have to view it as something you always do, not something you start when you need to fill a role immediately. There’s a fair amount of unpredictability in the process; if you find someone great for a role you won’t need for two months, you should still hire her now.

*Fire fast.

I have never met a newbie founder that fires fast enough; I have also never met a founder who doesn’t learn this lesson after a few years.

You will not get 100% of your hires right. When it’s obviously not working, it’s unlikely to start working. It’s better for everyone involved to part ways quickly, instead of hanging on to unrealistic dreams that it’s going to get better. This is especially true for the person you have to let go—if they’re just at your company for a couple of months, it’s a non-issue in future interviews. And everyone else at the company is probably aware that the person is not working out before you are.

Having to fire people is one of the worst things a founder has to do, but you have to just get it over with and trust that it will work out better than dragging things out.

*Put a little bit of rigor around the hiring process.

Make everyone on your team commit to a hire/no hire decision for everyone they meet, and write up their thoughts. If you get it wrong, this is useful to look back at later. It’s good to have a brief in-person discussion with the entire interviewing team after a candidate leaves.

Have someone take the candidate out to lunch or dinner. Insist that everyone is on time and prepared for interviews/auditions. Make sure every candidate leaves with a positive impression of your company.

Be organized—one person should coordinate the entire interview process, make sure every topic you want to cover gets covered, convene people for the discussion after all interviews are done, etc. Also, have a consistent framework for how you decide whether or not to hire—do you need unanimous consent?

Remember that despite being great at what they do your team may not be great interviewers. It’s important to teach people how to interview.

*Don’t hire.

Many founders hire just because it seems like a cool thing to do, and people always ask how many employees you have. Companies generally work better when they are smaller. It’s
always worth spending time to think about the least amount of projects/work you can feasibly do, and then having as small a team as possible to do it.

Don't hire for the sake of hiring. Hire because there is no other way to do what you want to do.

Good luck. Hiring is very hard but very important work. And don't forget that after you hire people, you need to keep them.

Remember to check in with people, be a good manager, have regular all hands meetings, make sure people are happy and challenged, etc. Always keep a sense of momentum at your company—that's important to retaining talent. Give people new roles every six months or so. And of course, continue to focus on bringing talented people into the company—that alone will make other good people want to stay.

Always be identifying and promoting new talent. This is not as sexy as thinking about new problems to solve, but it will make you successful.

Thanks to Patrick Collison, Andrew Mason, Keith Rabois, Geoff Ralston, and Nick Sivo for reading drafts of this.
12. Building for the enterprise
Lecture 12: Building for the Enterprise

Aaron Levie

Can we keep playing, yeah.

[Eye of the Tiger starts]

Can we turn it up a little bit so it has more pump up? Okay, there we go. Okay. Guys, we have to find the beat then clap to the beat. Okay, please stop the music. Please put on the presentation. Thank you. That will be about the most pumped up thing that happens in enterprise software. The rest is downhill from here. Thank you for that well rehearsed intro.

I'm Aaron Levie, CEO and co-founder of Box. Welcome to this edition on how to build an enterprise software company. This is my understanding, this is the course you are taking? Is that correct? No. So this is my job today. I am going to try and convince you that everyone else that speaks during this whole class is wrong and that you actually want to build an enterprise software company. Hopefully we will be able to work through this and you'll have a good sense of why it's super cool to be in enterprise. And why the perceptions of going into the consumer space, why it's so much fun are wrong, and why you want to go into enterprise software. Who wants to build an enterprise software company? Good, alright. Thank you very much. Hopefully we will do a vote at the end and hopefully that will not have shrunk. That's really the only goal I have today.

So we are going to talk about three things today. The first is the quick background of Box. Because when we first started out, we did not know we wanted to do enterprise software. So I want to go a little bit into why we went after enterprise and what we do today. Then we are going to talk the major factors that changed in enterprise software that make it possible to do a startup today. And finally we are going to look at patterns that are ways to recognize and go build a startup by yourself. Hopefully, that will be some practical, useful advice. Just as a forewarning, my voice, I've been speaking a lot the past few days. So hopefully I will be able to get to that third part of advice and make it.

Building for the enterprise, these are high level stats of Box. We have about two hundred forty thousand businesses that use Box, there are over twenty-seven million users that have brought Box into their organization, and ninety-nine percent of Fortune 500s. Actually that one percent is really Microsoft and they don't seem to want to buy from us. We have to work on that a little bit. A lot of users to bring Box into workplace environments, these are some of the organizations that are using the product. We have a very wide range of industries from manufacturing consumer products to companies like General Electric. Stanford Health Care actually uses the product for collaboration inside the medical department search. Between health care, media, manufacturing, these are some of the range of industries we serve.

So the question is, how did we get here? Because we didn't start the company to be an enterprise software company even though that is how things ended up happening. We launched the company in 2005, we got the idea back in college which was 2004. Was anybody using the internet back in 2004? Okay great. I didn't know if millennials used the internet or not, so great. Sorry, okay no more age jokes. Okay here is the point. So back in 2004, you might remember there wasn't a lot to do back then. It was boring right? This was before Facebook, this is certainly before Snapchat, so before much to do. You couldn't send people fifteen second messages or photos that disappeared because you didn't even...
have phones. So on the internet, in 2004, there wasn't a lot going on. This is sort of what the internet looked like, a barren deserted landscape. Just to clarify, the happy camel is Google, the sad camel is Yahoo! This is the internet in 2004. Yahoo! has done a lot better since then, but back in the mid 2004, they were trying to find their way. And Google was taking over the world. But this was the extent of the entire world.

So what we noticed in 2004, in college, was for some reason it was really hard to share files. And as simple of an idea as that is now, and you go back ten years. It was either really expensive or really hard to move data around through corporate companies. I had an internship at the time, most of my job using data was to copy printed out papers and put them in cabinets. That's what you do as an intern if you are not a computer science guy. So I was really really good at copying paper, unfortunately not a skill really used today. But it was really hard to share files. In classroom environments, you were working in large groups, it was also hard to share files. I went to USC, and USC gave you fifty megabytes of storage space. Fifty megabytes, you can basically store one file. Then it would auto delete every six months. So whoever was running IT at the time, they certainly weren't running hard drives. And so it was really, really hard to store and share files. Well, why don't we make it easy to store and share files from anywhere?

So we got the idea for, at the time, Box.net. And what we noticed was, there were a lot of factors that changed in the software world. The first was the cost of storage was dropping dramatically. So in our business, basically every year or two you could double the amount of storage and data goes into a hard drive. So what was uneconomical now because feasible. The cost of computing, the cost of storage has dropped. We had more powerful browsers, and networks. Firefox was just emerging. People were using the much faster internet for homes and the classroom. Then people had more locations that they wanted to store and share information with. So we had these three factors that were sort of emerging. So pull these factors back when I give some tactical advice. The first point to remember, always look for the changing technology factors. Every market that has a significant change in underlying were enabling factors was in an environment that was about to change in a very significant way. We were very fortunate in the need for data in the Cloud, was growing in importance. The cost and feasibility, was also not improving rapidly. We decided to put together this really quick version of Box and launched it as Box.net. The idea was, let's make it really easy to share files. It turned out the idea clicked. We got angel funding from this guy named Mark Cuban. This was before Shark Tank but it was very similar. So we got this funding and thought this was going to be super exciting.

We are going to drop out of college, we are going to move to the Bay area, and it's going to be awesome. And when you drop out of college, anybody drop out of college yet? Okay, good. Stay in school! When you drop out of college, everyone pictures it like it's going to be incredible. Bill Gates dropped out of college, it will be like Bill Gates. Or Michael Dell dropped out of college, it will be super exciting like Michael Dell. Steve Jobs dropped out of college, so this is what people imagine, but nobody ever remembers that this guy dropped out of college also.

So it's not really a guarantee that it's going to be successful. It's funny, I don't even know if this guy dropped out of college. It just seems like he had to. And I apologize if anyone is related to him, it's just a funny picture on the internet. So basically we decided we would drop out of college, we moved up to first Berkeley, then Palo Alto. We decided we were going to open up the product for free. We got hundreds of thousands of people to sign up for the product every single month. If you go to Box.net you get one free gigabyte of file storage space. Which once again was big back in 2006. But we were getting so many users, we were trying to figure out what to do.

What we ran into was a common problem that, really, any startup runs into. Really pronounced by our business model which was, for consumers we built a very robust, very reliable enterprise. We really
brought a really insignificant product. So for consumers, what we were running into was we had all these features you could pay for but a lot of consumers didn't need all those features. And for enterprises, we really didn't have enough securities and we didn't have enough capabilities around how enterprises want to use their data. So we had more than what a consumer needed and not enough that an enterprise needed. So we found ourselves at the juncture. We found ourselves basically in this period where its very difficult to figure out what we wanted to do with the business. So we had to make this choice. We were at this path, where we had to choose which path to go down.

This is back in early, mid 2006 up to late 2006. I was 23 at the time, my cofounder was 22. Our founding team was even younger, we had all dropped out of college. So in 2006, 2007, we imagined these two paths and the worlds were very very different. When you do a consumer startup it's basically lots of fun. You have parties all the time, it's just super exciting. Then in the enterprise you are battling these large, it's a rather thankless model because people just generally hate enterprise software. So that was sort of how we imagined the two paths, was we had to choose one of these two worlds. So we looked at that and thought. Okay consumer looks really fun, enterprise looks really hard and there is a lot of competition. At the same time, in this consumer space you are always fighting this issue of how do you monetize? How do you actually get people to pay for product? In the consumer space there are really only two business models that you can do.

You can either have people pay for your application or you could provide advertising on the application. To give you a little bit of perspective, these are today's numbers. In the consumer world there are about thirty-five billion spent on mobile apps every year. Pretty big number right? Thirty-five billion dollars. That's a lot of money being spent on mobile apps today. For advertising, the global digital advertising is $135 billion dollars. So most consumer businesses are going after, if you are not doing e-commerce, are going after about $170 billion dollars of either purchasing power on applications or global advertising around these types of services. So big number, a lot of opportunity there.

However, in the enterprise there are $3.7 trillion on enterprise IT every single year. These are the servers, the infrastructure, the software, the networking, the services. All of that stack of technology equates to a few trillion spent every year. What we realized was there was a rather wide delta between these two markets. We are going to be fighting to get consumers to pay a few dollars a month. And Google, Microsoft, and Apple will try to make this product free over time. And there were rumors that google drive was coming out. And all these products that were going to happen, are coming out. But in enterprise, its not about them trying to save money on IT. They are either trying to increase productivity, they are trying to increase business. So the value equation is very different. So the consumer, we have limited amount of money that we wish to conserve for as few things as possible that we are going to spend. On the enterprise it's a little bit of a shift, actually what can I get out of technology? How much value is that for me? So that was a really important data point.

However, the problem was that enterprise software was very unsexy right? Very competitive, very hard to build a business. It wasn't something you shot out of bed in the morning saying, I'm super excited to build an enterprise software company. And the reason for that was actually very straight forward at the time. The way that you built software was very slow. It was very slow because you couldn't break anything for customers, the sales process was very slow because customers take a long time to purchase technology. So I think everyone is used to this philosophy that when you are trying to sell enterprise software, it could take up to years for them to actually just buy the software. Then it could take even more years for them to implement the technology in the first place. So a lot of companies are around for years without their technology even used in the first place.

That felt like a huge problem, and not something that we wanted to be a part of. The technology itself is complex, I don't know how many people have had to use enterprise software but it's generally really
complicated. You try to figure out, why in God’s creation did a designer try to put forty-seven buttons on one page. You just can’t even understand it and the reason is something we will get into in a second. But basically there is just no love or care for the design or user service. The software is just complex. And finally, if that wasn’t bad enough, you have to figure out how you are going to sell this software. For anyone who loves the power of the internet, this notion of having a sales intermediary to get to your customer, seemed really unappealing. You have to hire a bunch of people, who are going to be in every country, they are going to be the only interface you have to your customer. You hire this guy named Chuck, and Chuck is going to roll in with a brief case and he is going to try to sell lots of enterprise software to the customer. Just so we are clear, this is what Chuck looks like. And that was the sale process that you, in the enterprise at least, that we imagined in our head. Chuck looks like a happy guy, but he is still an intermediary to getting your software. Why cant use the power of to internet and get our technology out here that way?

Why should we have to go through this sales intermediary as we scale up the business? I will get into it in a minute why we were wrong about the sales business. But this was sort of the fear that we had. And if that wasn’t hard enough, we had investors saying, in 2007, basically there is no way you are going to make it in enterprise. You again, are basically a founding team of 20 year olds. You don’t have anyone on your team that has been in an enterprise. Microsoft, Oracle, IBM, these companies are going to stomp on you. This is going to be very very hard to succeed. And to be fair they were right on several areas. We were a very inexperienced team. We were still very early in our careers. Our co-founder, for instance, looked like he was 13 years old. Just to be clear on what he looked like. So it sort of made sense right? This is him as our CFO, I think this is him at 29. But it looked like we were going to run off with the money and go to Disneyland. I appreciate why they didn't think we could pull it off. I can't imagine giving him money.

So, we decided that we still have to go do it. We have to give this our best shot. We are going to take the scale, the consumer experience, the DNA of our company and we are going to see if we can bring this into the enterprise. We were really fortunate. We had an investor, early in his career, make a belief on us because there was something changing with the enterprise that we would be able to take advantage of. We decided, if we are going to do the enterprise, if we were going to go after the enterprise, we were going to have to play with a very different set of rules. So what about the complexity of software can change in this era? What about the sales process is very slow can change in this new era? How do we move and go directly to the user the customer, instead of having this really indirect process at getting our technology out there? How do we build a design for the user instead of just for the RFP process that a customer is going to go through? So we looked at all of the factors that are true with the enterprise and we are going to do, not in all cases, the opposite. We are going to find what has changed in the technology world that we can build a newer, and better software company. That was the decision we embarked on, the path we embarked on 8 years ago. And that is why we have been focused on enterprise.

Today, again we have about two hundred and forty businesses using the product. And the reason is we architected the business model, we architected the software, we architected the solution to work in one specific version of the world, and it turned out that one solution was the one that happened. And I will go a little bit into what has changed in the world that we sort of built our company around. And what I would highly recommend to you, if you are building an enterprise software company to orient to your technology. So that was sort of why we made the decision, how we started to take on the problem.

So everything about enterprise, and by definition the software that the enterprise uses, has changed just in the past 5 years. If there ever was a magical time to build an enterprise software company, now is that time in terms of how much has changed in what is going on with organizations. Let's go over a couple of these things. The first is that most application companies are moving to the cloud. And the
biggest thing is, if you are going to start a business management company, or a business intelligence company, even a contact management company years ago you had to have your idea implemented in every single customer location. No matter how many customers you sold to, no matter what region you were in, every customer had to put that in their datacenter. That was the flaw with on premise computing. You were doing all this work, you were creating so much redundancy, it was the slowing down the whole process of delivering and building software for the enterprise. All of a sudden the cloud came around, things like Sales Source.com, things like Amazon Web Services, basically said. Why is it that every customer that wants to implement a couple servers, have to implement servers, put them in their data server, but security or networking around them, six months later they go live and a developer can use them in the organization, same thing with an application? They said, why does that make any sense today? We could just put together tens of thousands of servers, put them on demand, and you can use whatever you want, when you want and we can do that. That obviously is the definition of Cloud Computing. What's happening is CIO's and large enterprises are taking advantage of this. So it seems obvious to everyone in this room because you would never build your company by buying your own servers. You would start on google, yahoo, or ashore rather. But to an enterprise there are decades of infrastructure that now has to move to that cloud. So that's a massive shift that is actually happening.

We are moving to a world of cheaper, on demand computing from a world of expensive computing. The benefit of starting a startup is the customers don't have the same friction, they are going to go and adopt new technology. As soon as the computing because cheaper, its easier to adopt new solutions. Which means, their barrier for showing you in--the barrier is a lot lower which is great for startups. You are going from a world of customized platforms to standardize softwares. It use to be that you had to build all the customizations, all the customer experiences on top of the software itself and now customers are realizing that they won't open platforms and they can customize a layer on top of the product. It use to be that when you started an enterprise software, you could only sell to the top five or ten thousand companies in the world. Because only those companies had the ware with all, the talent, the infrastructure, and the budget to employ you technology into the enterprise. Today, literally a two person company can sign up for box, as well as we work with General Electric who has over 300 thousand employees. So the fact that you can now serve a small business anywhere in the world, as well as some of the largest on the planet means there are much larger markets you can go after. Which makes it an even better economical proposition to go after the enterprise. The platforms themselves are becoming more global. Our customers were internationally a couple weeks after starting the company. If you would have done enterprise the traditional way that would have take years to actually be able to go internally.

And finally, the most profound shift of all, mobile devices. iPhones, iPads, Androids, Tablets, IT of these models have become a lot more user led. It's fundamentally important. In an IT world, incumbents generally win because they have the existing relationship with the IT organization, with the CIO, with the spending power of that company. In a user lead model, users are bringing in their own technology. They're bring it in in the sales team, they are bringing it in in the marketing team, they are brining it in in finance and you can build software around that user. Which means they can bring the enterprise in and you can sell to the enterprise when they want to have better control, better security, better scalability.

So you still have the same model as a business software company but the way to get into the company now is through the end user. So those are quantitative factory changes. Just a couple quantitative changes, there is over nearly 2 billion smart phones on the planet. That changes every single IT model planet. Because it use to be 10 years ago, if you were managing technology for the company, you just had to manage the computers network that was inside your building. But now with billions of smart phones you have to manage ways of computing anywhere at any time on any network. And that becomes big in software companies, because no incumbent has built a technology stack that powers
Lecture 12: Building for the Enterprise

this line of work and how enterprises are using their data. So that creates a massive start up opportunity.

There are nearly 3 billion people online. That means that every single enterprise is changing how they are going to give their own products to their customers. Which means that every industry changes. There are only two times, two moments of opportunity where a technology revolution will happen in an enterprise. The first is where raw materials change. So cost of computing goes down and they centralize and let people use it on demand. The second thing that can change, is the very people that these enterprises have to go after need new experiences at that enterprises product. Let me give you an example. If you go off campus you probably use something like Uber or Lyft. If you are in the shipping business, if you are in the transportation business, Uber represents a massive change to your industry. So you can't let Uber exist without understanding, what are the implications of Uber? What are the implications of Instacart? What are the implications of Lyft to my business model?

So in a world where enterprises are dealing with that kind of change, you are going to need new technology to help them create their business models, how they adapt to this disruption. This is why it's such an amazing time to even start vertical software companies for industries. Right now every single industry is going through a business model and technology orientated disruption. Means they are going to need technology from start ups to help them work through this. I will give you a couple of examples: so in the retail industry there's this vision of omni channel or multichannel commerce. You are going to shop online, you are going to shop on your phone, you are going to shop in a store, and you want things to be delivered to you as well. So most of the incumbent technology does not power multichannel commerce. No one is prepared, what does it mean when consumers want to go buy goods anytime from anywhere with better information, better intelligence. So every retailer in the world is going to need a new technology stack to power their retail experiences.

In the healthcare space, every single health care institution is trying to find ways of building more personalized experiences, more predictive experiences, they want to have medicine be adapted to the individual. As the business model of healthcare changes to being about the surgery, charging for the check up, and instead, really where the customer pays for the wellness and staying healthy. Then all of a sudden every healthcare institution needs technology to deliver health care experiences. They are going to want to deliver telemedicine. They are going to want to deliver health care in more regional locations instead of just in the monolithic hospital environment. There are going to be new use cases coming around. How are our healthcare providers get connected to one another so one doctor can make better decisions? All of these things are going to require new enterprise software to power these businesses and industries.

In the media space, as an example, you have a world where the industry is going from really linear programming, whether that's television or that's music or that's movies, it's a linear supply chain oriented business model, when a film gets made, it goes to the theater for 3 months, then afterwards it goes to iTunes or other platforms, to a world where people want experiences on demand. So that's going to change how distribution works on a scale of 3 billion people on the internet, and again no media network has a platform that is actually going to power how content, data, and information moves to this system at scale. I was just in LA yesterday, meeting with a media company that has basically done predictive analytics to find their potential moviegoers in the middle of 3 billion users. They want to be hyper targeted on how to get to the specific 3 billion fans that are into a certain film types. And so all of a sudden you have a movie company who needs big data and they need business intelligence and marketing in order to go and market and distribute.

This is where two industries come together, where all new software is going to be necessary. So every industry is going through some form of this change. You can take any industry you want and zoom into
it and say, what are the underlying technology factors that are going to change the business model for the next couple of years? And then there is going to need to be software to power those types of experiences. Think of the future of water, who is going to power that? That's going to need software, I'm sure.

So basically every company in the world, the great thing about being at Stanford is we study the technology. And we think of the technology industry as an industry. But in reality what is happening, is every industry is going to have a technology component of what they do. Enterprises are not going to be able to survive in the future if they do not get good at technology. If they don't have competency at leveraging data and using these new tools. But they are going to do that by working with what we call the technology industry.

Instead of everyone else building out in these expertise for themselves. So there's going to be a lot of partnership over the next five to ten years where companies are going to need technology to work smarter, to work faster, they are going to need to do this more securely. And this is not only going to change how individuals work in these environments but ultimately change the business models of these companies. So that was chapter two of this.

That is all the things that have changed. Now some practical advice to help you get started. And to be far, most of this advice is looking through the lens of retrospect. Which means this is not how this is going to happen, but I can look back through time and say that these are the things that led this to be true. So it's hard to be super deterministic about building a company. You may not have all these things figured out. But this will give you a sense of pattern to recognize as you are building, or thinking about building an enterprise software company.

So the first one is spot technology disruptions. This is going to be true whether you are building consumer or enterprise. The rest are more enterprise, but this is just fundamental if you are going to build a tech company. You have to look for new enabling technologies, or major trends, like fundamental trends, that create a wide gap between how things are done and how they can be done. Looking back in time to our business, the gap was basically storage was getting cheaper, internet was getting faster, browsers where getting better yet we are still sharing files with this very complicated, very cumbersome means. Anytime, between the delta of what is possible, and how things work today is at its widest. That is an opportunity to build new technology to go solve a problem. As you are looking at the enterprise, the question is, what about the cost of computing dropping so rapidly changes what enterprises can do with their data?

What does it do to change from a business standpoint? What was impossible, because of either economic feasibility or technical feasibility that 10-15 years ago is now possible. A fun thing to do every now and then, if you find a newspaper article from the 1990's or 1980's, business articles about technology, all we are really doing is repeating all the technologies we tried 10, 20, 30 years ago. It was too expensive, too unusable, and we didn't have the enabling technologies to make it possible. You can see this concept emerge, something that was impossible 5, 10 years ago is now very practical. I will give you an example. There is a company called PlanGrid, does anyone know what PlanGrid does? Okay, cool. Are you in the construction industry? You are? Oh my god! What does that even mean?

Q: PlanGrid? Or construction?

Aaron Levie: Construction.

Q: I work at a job site, we build buildings.
Aaron Levie: Holy crap, that's great. Basically PlanGrid is a mobile application that lets you manage construction projects, lets you access your blueprints, lets you manage all the data around a construction process. And what this company realized is, 4 million dollars, I think, are spent every year printing out blueprints. And they have all the prints and updates to them anytime there a change, then they have to ripple and cascade through a very wide network of contractors and construction workers every time. Even if it's one slight, minute change, suddenly they realized, with the iPad, we have the perfect form factor to load up blueprints and content. This is something that can ripple through the construction industry, which isn't really known for high technology, except on the design side. How can they build technology that makes data collaboration problem really seamless and easy to do in an industry that hasn't really changed in a while? It was a perfect discovery of a change in a market and figuring out how those two things converge. Then this team built a great startup for it that is doing incredibly well and taking over the construction industry as proven by this individual. Thank you.

The next thing is, in enterprise, you want to start intentionally small. What I mean by that is you want to find, this is more true with all companies in an enterprise in a user light paradigm, you want to find the wedge that is sort of natural that you can create a product that will slip in the gaps of other existing products. But something that you think over time expands to be a more important product of the enterprise structure. What you want to start to do is say, we will take this sliver of a problem, we are going to make the user experience on the incredible. We are either going to change the business model, we are going to create new technology to make this previously problem really really simple. It might feel small at first, maybe you are going after small businesses and then you are going to go up market. Maybe you are starting with a sliver of the use case and expand out, but you intentionally start small. Because you will not be able to compete with an incumbent because the incumbent is always going to go for the full solution. So you have to find, what are the gaps in the full solution, that are significant enough that the customer is going to want to solve the problem with a discreet technology. But over time you are going to be able to expand. Again to either larger customers or to more use cases over time.

Great example is ZenPayroll. ZenPayroll was started by Stanford graduates couple of years ago. Basically, they discovered that the payroll is some small business is complicated and incredibly annoying process. That is because we use the same vendors that we have for decades to do that and they were digitally ordinated. You didn't get your payments as a receipt over email. Very complicated. You didn't get to see graphs of your salaries. There was really no good data around this. And they said we are going to take off the slice that is most painful to start out at, around hiring people and paying people. Just that payroll management process. We are going to plug into a lot of existing structure. But we are going to make it dead symbol to go do this. And now they are able to move up market over time as well as deliver new services over time. And what happens is the incumbent in this market, eventually looks at something like Zen Payroll and says, well thats small. Its only for small businesses'. How can it be very powerful? But thats just the start. As they get that wedge, as they fit into the market, they are going to be able to expand again over time. Build out more services and more capabilities. But they found just the right, exact opening to build a new company and have the emerge.

Then next you really want to find asymmetries. You want to do things that incumbents can't or won't do because either the economics don't make sense for them, the economics are so unusual, or because technically they can't. I will give you two examples. So, if you are going to build software today for the enterprise that goes after an incumbent category, that has more of a sweet oriented approach. Then what you are going to want to do is build technology that is platform agnostic. What sweet players will do is want everything to be integrated with itself, and theres more value with the vertical integration. But you want to go after a different access. Which you want your technology to work across all the platforms. That way you can work with so many different kinds of customers. You can be an ally to so many kinds of platforms, which a traditional incumbent is not able to do. That is technically infeasible.
because its architecture or its fundamental component business model to not do that. The other thing is, trying to do things that is economic feasible. You can look at the cost structure of an incumbent company and discover where they are not going to be able to drop their prices. Because that business model is fundamental to the company. Or where can you find ways of monetizing the customer that are unusual or unique that no one has discovered before, thus making impractical for anyone else to do.

There is a company called Zenefits where they have an HR management software company that helps you as a small startup manage all your benefits, all your HR information. And instead of charging the startup that may not value the software stage they are at, they realize they can get commission from the insurance companies that pays for the ability to use their software. The customer itself is not paying for Zenefits. Zenefits platform is being paid for by the insurance company and they have thus created a business model that no other software company has been able to think of or attack. And they are equally going and disrupting a category that has not seen a lot of innovation previously which is the health and benefits space in small businesses.

The next is you want to find the mostly crazy, but still reasonable outliers within the customer ecosystem. So you need to find the customers that are at the edge of the business, their business model, their industry and find the unique characteristics of those customers. Leverage them as your early adopters. Paul Graham has a great article where he talks about living in the future and building what is missing when you are living in the future. Thats an easy way to spot trends and patterns about disruption that is playing out. The same is true in the workplace. If you find customers that are working in the future, you will be able to work with them to find what is missing in the future. And how do we build technology that supports all these new use cases that are going to emerge? There is a company called Skycatch that does enterprise drones. At first it seems kind of weird, but in construction space, farming they are using drones now for data capture and modeling different environments. So this company is able to find all the companies that are on the bleeding edge of their industry. What is unique, or new about how those businesses operate. And they worked with a lot of those early adopters to establish their platform. Which really is first enterprise drone company. So the idea is, go look at your market. Find the customers on the bleeding edge of their market who use technology to get a head. And that use technology for performance advantages, and go work with them to see how your product can evolve.

Listen to your customers but don't always build exactly what they are telling you. This is a really key distinction around building enterprise software. Your customers are going to have a large number of requests. Your job is to instill those lists down into the ultimate product. This does not mean that you are not going to build exactly what they tell you to build. It is your job to listen to their problems, and translate those into what is going to build the best and simplest solution for them. It's really your job, and Palantir does a really good example of very very complex issues and then scaling them down into simple solutions for complex problems that the customer would not have known how to ask for.

You want to modularize not customize. So build a platform as opposed to building all the custom technology and customer vertical experiences into the software itself. Make sure you really think about openness and APIs as a way of building experiences. Don't build that directly into the product. Focus on the user always. The magical thing about building an enterprise software company right now is you can keep consumer information at the center of the product. That will always mean that adoption is easier, that your product has a much better chance of going viral. It becomes easier to sell in the organization. Always make sure you bring consumer DNA into the product. Your product should sell itself. But that does not mean you don't need sales people. So this is a really important distinction. Leverage everything about the internet, leverage everything about users to get to your customers. But you still will likely need sales as a way to help your customers navigate your product, help your customers navigate the competitive landscape and ecosystem. So you are going to want very domain specific sales
associates that are going to be helpful for your customer in deploying enabled in these positions. But don't make that be a substitute, don't make that be a handicap for not building a great product. So you fundamentally build a product inside that. A company called Mixpanel comes in through the developer and eventually sells to that organization with a more inside sales process. Also read these three books: Crossing the Chasm, the Innovators Dilemma, and Behind the Cloud. These three combined, if you binge and read them all, you will come out ahead.

So in closing, today, right now is an amazing time to start a software company. I wish you the best of luck. If it doesn't work, we are hiring. The only other thing is, please do not compete with me because I have a lot of competition already. Ideally either come work with us or build your own company.

So thank you very much!
Lecture 12: Building for the Enterprise

Lecture 12: The Continuous Productivity of Aaron Levie


Ted Greenwald

Aaron Levie bounds onstage with the swagger of a standup comic. But he's not performing at the Comedy Store. He's in the Grand Ballroom at San Francisco's Hilton Union Square kicking off BoxWorks, his company's annual customer conference. Steve Jobs had his black turtleneck, Mark Zuckerberg has his gray hoodie; Levie's uniform is a staid black suit, a capitulation to the buttoned-down enterprise software market he aims to conquer. But he spices it up with a cheeky pair of colorful sneakers. Today they're bright red.

First order of business: the choice of one of his favorite bands, Blink 182, to close Box's two-day event. "We wanted to engage a younger demographic, so the first choice was Miley Cyrus," he says, calmly pacing the stage. "But in her contract, she stipulated that we needed to call the conference BoxTwerks." A chuckle ripples through the crowd. "Don't worry," he adds. "The jokes will get better."

They do. He roasts competitors like Microsoft (if he were considered to fill Redmond's newly empty CEO slot, would he have to fix the company or just get a new version of Windows out the door?) and industry icons like Larry Ellison (if New Zealand beats the Oracle CEO's boat in the America's Cup race, Ellison could simply acquire the country and shut it down). He even pokes a little fun at himself, showing a goofy picture of what he calls Box's entry in the next America's Cup: Levie pedaling a paddleboat across San Francisco Bay.

It's a lighthearted performance, but Levie, 28, takes his business seriously. He wants to provide the Internet with something fundamental: a storage system for business-related files that employees can access on any device. In his view, Box's technology is the infrastructure for a new way of working that's more spontaneous, fluid, collaborative, and productive.

That aspiration places Box between the enterprise software equivalents of Scylla and Charybdis. On one side is Microsoft, still a formidable force in the business software market. On the other is Dropbox, a phenomenally popular consumer-focused service that sneaks past corporate gatekeepers tucked inside employees' smartphones. And yet Box may do far more than either rival to virtualize the office.

The forces propelling Box have been gathering for decades. When mainframe computers gave way to PCs, large companies stocked up on packaged software from companies like Microsoft and Oracle. To run it, they invested in racks of servers, fleets of desktop PCs, and armies of information technology managers. Then along came the Internet. Programs like Salesforce offered software as a service, eliminating packaged software, automating updates, and saving infrastructure and management overhead by running in the cloud. With the rise of mobile devices, employees brought their personal devices into the office, packed with their own apps that routed around management-sanctioned software—a phenomenon encapsulated by the phrase "the consumerization of IT." The traditional corporate IT department began to appear obsolete.

Along the way, IT managers lost control over one of a company's most valuable assets: documents. If employees use their own e-mail accounts to share secret contracts or store presentations about upcoming products in a consumer-grade file storage service, there's a risk that the details could ricochet around the blogosphere in minutes.

Levie has designed Box to put the IT department back in control, to the delight of customers including Amazon, GlaxoSmithKline, Procter & Gamble, Siemens, and Toyota—97 percent of the Fortune 500, as he's fond of saying. Like a number of similar services, Box provides file storage in the cloud—remote data centers somewhere on the Internet. It's simple enough for individuals to get up and running on their own at little or no cost. Users access the service from Box's website, its mobile app, or software running on a PC. Move a file into Box, and the file becomes available on many devices; change the file, and the alterations propagate to the other devices as well. But beneath the surface, Box provides features like security and permissions control that let corporate IT departments manage the way information flows through organizations. To get these professional-grade features, companies pay Box between $5 and $35 monthly for every employee who uses the system.

Box has 20 million users. That's few compared with Microsoft, which holds more than 385 million accounts between its consumer- and business-focused file storage services, SkyDrive and SharePoint. It's also puny next to Dropbox, with 200 million accounts. Even so, Box has advantages over both in the corporate market. Largely
Aaron Levie has never taken much interest in leisure. Born in Boulder, Colorado, he was pulling weeds and walking neighbors’ dogs for money by the time he was eight years old. When he was 10, his family moved to Mercer Island, a strip of land in Lake Washington between Seattle and Bellevue, a 20-minute drive from Microsoft’s headquarters. The tech bubble was beginning to inflate; he and his parents, a chemical engineer and a speech pathologist, discussed business ideas around the dinner table. He was an indifferent student, but he spent his free time building websites: a search engine, a real-estate site, a downloadable toolbar that pushed news. (“It probably gave you a virus,” he jokes.) His friend Jeff Queisser, now Box’s vice president of technical operations, supplied technical know-how. “About every month, I’d get a call at 1 a.m. to come to his hot tub, where he’d pitch an idea,” Queisser recalls.

Levie wanted to be a movie director in the mold of Quentin Tarantino, but the University of Southern California’s film school rejected his application. He settled for USC’s Marshall School of Business. During his sophomore year in 2004, a marketing class project led him to research online data storage. Early providers of that technology had been devastated when the dot-com bubble burst in 2001. Yet technology had evolved to the point where storing files on a hard drive in the cloud could be practical for mainstream computer users. “There was a disconnect between companies that existed and the size of the opportunity,” he says.

He roped in Dylan Smith, a Mercer Island friend who was studying economics at Duke University, to handle finance, and in April 2005 the pair launched Box on roughly $20,000 Smith had won at online poker. Within weeks, they had thousands of customers. Off to a heady start, they sent an e-mail to the billionaire Mark Cuban, whose popular blog, they thought, could boost their public profile. Cuban responded with a request to invest. The founders gladly cashed his $350,000 check, dropped out of college, and moved into Levie’s uncle’s garage in Berkeley.

By 2007, Box’s user base had doubled 20 times over and annual revenue was around $1 million. But Levie felt uneasy. The price of hard disks was falling 50 percent every 12 to 18 months. As online storage became a commodity, what would stop Apple, Google, or Microsoft from giving it to customers free? He noticed that the customers who stuck around longest weren’t storing MP3s or JPEGs but Word,

written a decade ago, Microsoft’s code is intricately entwined with a pre-mobile, desktop-based, intranet-bound way of organizing corporate IT. The company has been struggling to catch up with the rise of the cloud and mobile computing, while Box is designed to fit smoothly into an increasingly informal work culture born of easy-to-use Web and mobile apps. As for Dropbox, it has spent years catering to consumers and might well spend many more building enterprise-grade technology.

But Levie’s vision may be the decisive factor. Box doesn’t merely store documents, he points out, but facilitates communication around them. And communication—not a nicely formatted, ready-to-publish document—is the crucial product of work. The latest updates to Box’s service make document archives interactive, allowing users to add metadata, scroll rapidly through high-resolution previews, and search for snippets of text. The system is also taking a leap from content storage to content generation with the addition of Box Notes, a basic text editor that encourages collaboration: avatar icons skip across the screen in real time to show who’s typing what.

In this way, Levie threatens more than just other cloud storage providers. He’s shoveling coal into a locomotive of cloud-based enterprise services that promises to mow down any software company if it can’t translate its desktop offerings into sleek mobile apps that interact with their users’ data anytime, anywhere, on any device.

“The cloud is going to drive a new way of working,” he says after the conference. “The ability to deliver medical research from a lab to a doctor in seconds, or from an educational publisher to a student—it’s about real-time, collaborative, synchronous information sharing. It’s going to change work. Not just the technology of work, but work itself.”

The cloud—or, more precisely, the rigor of running a rapidly expanding cloud-based software company—has certainly shaped Levie’s routine. At 11 a.m., he arrives at Box’s office, a sprawling workspace with an Italianate exterior in Los Altos, California. He attends meetings until 6:30 p.m. or so, whereupon he’ll have another meeting over dinner or walk down El Camino Real to a Vietnamese pho house. After returning to the office, he naps for 20 minutes. Then he’s back on the job. He leaves at 2 a.m. and heads for the nearby apartment he shares with his longtime girlfriend, and he’s asleep by 3:30. By 10:15 a.m. he’s awake and ready to resume plotting his conquest of the workplace.

During the brief time between arriving at his apartment and hitting the pillow, he reads: manuals of business strategy, biographies of celebrated entrepreneurs, histories of iconic companies. “He has read more books about the tech industry than anyone I know,” says Josh Stein, an early champion of his at the VC firm Draper Fisher Jurvetson, one of the companies that have collectively invested more than $400 million in Box. Indeed, in conversation, much of the time Levie sounds less like a first-time entrepreneur than a professor lecturing on the latest theories of the technology adoption cycle.

These bedtime stories are also scary enough to keep Levie awake (and in the office) at night. “It creates this deep paranoia,” he says. “At any moment, you’re making decisions that might determine the survival of your company. That doesn’t lend itself to being in Hawaii for a month.”

Levie wanted to be a movie director in the mold of Quentin Tarantino, but the University of Southern California’s film school rejected his application. He settled for USC’s Marshall School of Business. During his sophomore year in 2004, a marketing class project led him to research online data storage. Early providers of that technology had been devastated when the dot-com bubble burst in 2001. Yet technology had evolved to the point where storing files on a hard drive in the cloud could be practical for mainstream computer users. “There was a disconnect between companies that existed and the size of the opportunity,” he says.

He roped in Dylan Smith, a Mercer Island friend who was studying economics at Duke University, to handle finance, and in April 2005 the pair launched Box on roughly $20,000 Smith had won at online poker. Within weeks, they had thousands of customers. Off to a heady start, they sent an e-mail to the billionaire Mark Cuban, whose popular blog, they thought, could boost their public profile. Cuban responded with a request to invest. The founders gladly cashed his $350,000 check, dropped out of college, and moved into Levie’s uncle’s garage in Berkeley.

By 2007, Box’s user base had doubled 20 times over and annual revenue was around $1 million. But Levie felt uneasy. The price of hard disks was falling 50 percent every 12 to 18 months. As online storage became a commodity, what would stop Apple, Google, or Microsoft from giving it to customers free? He noticed that the customers who stuck around longest weren’t storing MP3s or JPEGs but Word,
Excel, and PDF files. In other words, business customers. Moreover, their colleagues would follow their lead, generating a steady stream of new sign-ups. Levie decided to ditch the fickle consumer market and focus on serving enterprises, companies with thousands of employees, which would be willing to pay for a storage service tailored to their needs. He set about adding the capabilities required by large businesses: search, security, and the ability to create and delete accounts, manage file access, and grant permission to view, edit, or delete.

In embracing enterprise customers, Levie took on what was, at the time, the biggest tech company in the world: Microsoft. And Redmond might have crushed him but for a stroke of luck. In late 2007, Apple introduced the iPhone. For many people, the device was their first smartphone and the apps they downloaded transformed e-mail, document viewing, and even document editing into mobile experiences. Suddenly, employees were liberated from the strictly managed environment of corporate IT, with its-password-protected intranets and sluggish virtual private networks. If they found the office regime too restrictive, they simply downloaded apps that ran in the cloud—including one from Box.

As it happened, Apple, Google, and Microsoft did introduce consumer-grade file storage services in the cloud. Microsoft launched SkyDrive in 2007, to a collective yawn outside the desktop-bound world of Windows. Apple’s iCloud limped out in 2011, and Google Drive finally appeared in 2012, fully seven years after Box’s debut. Meanwhile, Dropbox launched in 2008 and quickly garnered rave reviews, a rapidly growing user base, and investments from top VCs. Today, it dominates the consumer market that Box abandoned.

But Levie never looked back.

Box’s office is a warren of desks, partitions, and meeting rooms with names like Watson (for IBM’s founder) and Revenue Bong (Levie’s off-the-cuff misremembering of the marketing phrase “sales funnel”). In the room called Fry’s (as in the electronics retailer), the CEO sits with eight colleagues around a long oak table. He’s wearing his black suit jacket over a bright turquoise T-shirt bearing the Box logo and a rainbow. It’s an odd combination, but it barely hints at the rest of his ensemble, hidden beneath the tabletop: neon-yellow shorts, calf-high turquoise socks (to match the shirt), and crimson sneakers. Today is National Coming Out Day, and the outfit is a show of solidarity.

With two cups of coffee on the table before him, Levie peers intently at the slides projected on the far wall. He drills the team, asking whether a given set of numbers are actual or projected and why the targets are so low. (“Five million for 2013? We should do 10. Let’s do 20!”) He swivels and tips his chair as he talks. Within a few minutes, the second cup is empty.

As the meeting winds to a close, Levie stands up, revealing his full Coming Out Day costume. “I’m going to jump out,” he says, and strides from the room on bare, caffeine-fueled legs. Moments later, a Box employee pokes his head in the door. “Aaron just ran by in a pair of yellow shorts,” he says. “Is everything okay?”

Levie opens the glass door of the pho house at 6:30 p.m. sharp and takes a booth. The waitress doesn’t even ask for his order; it’s always chicken soup, extra noodles, and a can of A&W root beer. Stirring his bowl, he explains that Box’s prospects depend on its ability to transform work from a serial march of e-mails, meetings, and reports to a parallel process called “continuous productivity.”

The phrase comes from, of all people, a former Microsoft executive—Steven Sinofsky, who at various times oversaw Windows, Office, and Internet Explorer, and left the company abruptly in late 2012 after the turbulent release of Windows 8. Levie saw the news and contacted him by poking him on Facebook. “Who does that anymore?” Sinofsky says. “I guess he thought I was an old person.” The two met over chicken pho with extra noodles, and Sinofsky soon joined Box as an advisor.

Sinofsky’s notion of continuous productivity goes like this: In traditional organizations, information is concentrated at the top of the management hierarchy and dispensed on a schedule. In connected, mobile organizations, on the other hand, every employee has equal access to information, potentially in real time as it accrues. This tends to flatten the
management hierarchy; the boss may call the shots, but they're readily redirected by employees. Moreover, workers can share information easily with people outside the company. This tends to dissolve organizational boundaries. The tempo of activity picks up, data replaces assumptions, and execution takes precedence over strategy.

Sinofsky's ideas reminded Levie of a 1937 essay entitled “The Nature of the Firm,” in which economist Ronald Coase laid out a rationale for why companies exist: they save the cost, in time and money, of organizing, disbanding, and reorganizing for every new project. “That was true in an era when we didn’t have common interfaces between organizations,” Levie explains. Not anymore. Increasingly, companies can assemble the resources they need on the fly: data centers for hire, contract manufacturing, crowdsourcing. More to the point, as the pace of change accelerates, they have no other choice.

Levie wants to put Box at the heart of this transformation. A key part of his plan is to add features and apps tailored to the needs of specific industries, including education, finance, government, health care, law, media, packaged goods, and retailing. Next, Levie envisions connecting not just companies but the industries themselves. To make a Hollywood movie, he points out, files must be shared among studios, agents, distributors, promoters, and lawyers. “At every point of sharing, there’s a slowdown,” he says. “The big question is how to accelerate that process.” His answer: by linking partners, suppliers, contractors, and so on to a synchronized collaboration service in the cloud.

A bigger question is whether businesses should surrender their information to a cloud service provider. Many find the cost savings compelling. But some competitors are betting that enterprises will need to keep files in-house, either because those files are extremely large—making them slow to upload, synchronize, and access online—or because they’re simply too sensitive to store on the public Internet. A company called Egnyte, for instance, offers a so-called hybrid solution that combines cloud and on-premises storage. Such an arrangement might appeal to anyone worried by revelations that the U.S. government—or other snoops—can plunder data held in the cloud.

Scripps Networks, which produces shows for cable TV, is an early explorer of this terra incognita. The company, which is based in Knoxville, Tennessee, and maintains offices in London, Rio de Janeiro, and Singapore, adopted Box after the CEO gave every senior executive an iPad in 2011 without informing the IT department. Scripps had been using SharePoint, but Microsoft's program didn’t support Apple devices at the time, and it proved unwieldy for ad hoc collaboration, says Chuck Hurst, VP of media and content distribution. Instead, employees were sharing confidential files through Dropbox and other systems that lacked enterprise administration capabilities. The legal department was having fits.

Hurst brought in Box in late 2012, and it has become integral to Scripps’s operations. The marketing department uses it to exchange assets with advertising agencies. The sales reps run presentations directly out of Box. “They can share things quickly and we don’t get in their way,” he says, “so they’re happy.” Box isn’t yet ready to take on the massive files required for production and broadcast video, but Hurst believes it will eventually. At that point, it could revolutionize the way things are done in his industry.

At the restaurant, Levie slurps up the last of his pho. Seven o’clock is only the middle of his workday. The office will be mostly empty when he returns, but that leaves him free to contemplate his next moves. “We’re only 1 percent of the way toward what’s possible in this space,” he says. Personal computers didn’t transform business until there was one on every desk, he points out.

Similarly, cloud computing won’t transform the way we work until every office across the world is using it. Meanwhile, people born in 2014 will never use a desktop or laptop. They’ll know only phones, tablets, Google Glass, and whatever comes next. “The PC shift affected millions; this will affect billions,” he says. “The opportunity is way larger than in previous eras of enterprise computing.”

With that, he pays the check and heads back to work, where the task of making an already always-on world spin ever faster, more efficiently, and more productively never ends.

Ted Greenwald is a freelance journalist in Silicon Valley who has written for Bloomberg BusinessWeek, Fortune, and Wired. He profiled Dwolla founder Ben Milne in the September/October 2013 issue.
Lecture 12: Robert Cialdini


Influence

Cialdini's theory of influence is based on the principles of reciprocity, commitment and consistency, social proof, authority, liking, and scarcity.

6 key principles of influence by Robert Cialdini

1. **Reciprocity** – People tend to return a favor, thus the pervasiveness of free samples in marketing. In his conferences, he often uses the example of Ethiopia providing thousands of dollars in humanitarian aid to Mexico just after the 1985 earthquake, despite Ethiopia suffering from a crippling famine and civil war at the time. Ethiopia had been reciprocating for the diplomatic support Mexico provided when Italy invaded Ethiopia in 1935. The good cop/bad cop strategy is also based on this principle.

2. **Commitment and Consistency** – If people commit, orally or in writing, to an idea or goal, they are more likely to honor that commitment because of establishing that idea or goal as being congruent with their self-image. Even if the original incentive or motivation is removed after they have already agreed, they will continue to honor the agreement. Cialdini notes Chinese brainwashing on American prisoners of war to rewrite their self-image and gain automatic unenforced compliance. See cognitive dissonance.

3. **Social Proof** – People will do things that they see other people are doing. For example, in one experiment, one or more confederates would look up into the sky; bystanders would then look up into the sky to see what they were seeing. At one point this experiment aborted, as so many people were looking up that they stopped traffic. See conformity, and the Asch conformity experiments.

4. **Authority** – People will tend to obey authority figures, even if they are asked to perform objectionable acts. Cialdini cites incidents such as the Milgram experiments in the early 1960s and the My Lai massacre.

5. **Liking** – People are easily persuaded by other people that they like. Cialdini cites the marketing of Tupperware in what might now be called viral marketing. People were more likely to buy if they liked the person selling it to them. Some of the many biases favoring more attractive people are discussed. See physical attractiveness stereotype.

6. **Scarcity** – Perceived scarcity will generate demand. For example, saying offers are available for a "limited time only" encourages sales. His seminal 1984 book, *Influence: The Psychology of Persuasion*, was based on three "undercover" years applying for and training at used car dealerships, fund-raising organizations, and telemarketing firms to observe real-life situations of persuasion. It's been mentioned in *50 Psychology Classics* (ISBN 978-1-85788-386-2) by Tom Butler-Bowdon and *Return On Influence* (McGraw-Hill 2012) by Mark W. Schaefer.
In doing research for a post on "The Enterprise Cool Kids" at the tail end of last year, I interviewed Silicon Valley veteran Marc Andreessen about where he thought the enterprise was headed.

While excerpts of that interview made it into the post, the transcript of the entire interview was so good it deserved to be published in its entirety.

Alexia Tsotsis: Since people like me (millennials) are putting pressure on our IT departments to buy products that we can actually use and aren't blinded by, what do you think the enterprise space will look like in the next five years?

Marc Andreessen: Yeah. So let me maybe start with sort of — top-down and bottoms-up is how we think about it, because both are important — so let me start with historical context and then maybe go to the stuff happening right now. Is that all set?

Alexia Tsotsis: Yeah, it's perfect.

Marc Andreessen: So the computer industry started in 1950 and basically ran for 50 years with the same model, which was a model where all of the new computers, all the new technology, all the new software started out being sold for the highest prices to the biggest organizations. So originally the customer was the Department of Defense. It was the first customer for the computer. In fact, one of the big first computers was called SAGE, which was a missile defense, the first missile-defense computer, which was like one of the first computers in the history of the world which got sold to the Department of Defense for, I don’t know, tens and tens of millions of dollars at the time. Maybe hundreds of millions of dollars in current dollars.

And then five years later computers became — they dropped half in price and then the big insurance companies could buy them, and that’s when Thomas Watson, who ran IBM at the time, was quoted as saying, “There’s only a market need in the world for five computers.”

The reason that wasn’t crazy when he said it is because there were only five organizations that were big enough to buy a computer. So that’s how it started. And then IBM came along and productized the mainframe, and then all of a sudden big normal companies — manufacturing companies and banks — could start to buy computers. And then DEC came along and came out with the minicomputer, and then all of a sudden smaller companies could start to buy computers. And then the PC came out and then all of a sudden smaller companies could start to buy computers. But the PC only ever got to hundreds of millions of people. It never got to billions of people.

Now, the smartphone has come out and it can get to billions of people.

And so it has always been this kind of trickle-down model for 50 years. We think that basically about 10 years ago the model flipped. And so we think that the model flipped to a model where, today, where the most interesting and advanced new technology now comes out for the consumer first. And then small businesses start to use it. And then medium-size businesses start to use it, and then large businesses start to use it, and then eventually the government starts to use it. But this is a complete change from the way it has always worked.

Alexia Tsotsis: It’s grassroots versus trickle-down.

Marc Andreessen: Versus trickle-down. And the reason is because — the reason fundamentally is because now that you have got these things, you have — now that you have a computer in everybody’s hand, all of a sudden all these barriers — it used to be these barriers to market entry were so big, it used to be there just weren’t that many early adopters in the world. To bring out a new technology for consumers first, you just had a very long road to go down to try to find people who actually would pay money for something.

And now all of a sudden you have got this global market of all these early adopters that have smartphones connected to the Internet, and they can just pick up their things and run with them.

And of course consumers can make buying decisions much more quickly than businesses can, because for the consumer, they either like it or they don’t, whereas businesses have to go through these long and involved processes.

So that’s the big, big, big change that’s happened. And that’s been reflected in the entrepreneurial community, where entrepreneurs, especially between 2000 and 2008, entrepreneurs really only wanted to do — for the most part wanted to do consumer software, because that’s the only software that they could actually get anybody to adopt. It became very hard to get businesses to adopt new stuff.

In the last five years, there’s been this sort of acknowledgment of the consumerization of the enterprise, which is consumer product development, design methods applied to business software, of which SaaS and cloud and all these things are examples. Salesforce.com, Evernote is an example. So now you have got the rise of this new set of companies that are sort of consumerized technology for businesses.

Then from a bottoms-up standpoint what you said is exactly right, I think, which is that the new generation of employees grew up on smartphones and tablets and touch and everything, social networking and Twitter and everything else. And so if you take a typical mainframe or, even these days, PC-based system and you give it to a 22-year-old college graduate, it’s like beaming in products out of the...
Stone Age. Why would you do that? Why would you force people to use all this old stuff? And then that leads to the big thing that’s starting to happen right now, which is this “Bring Your Own Device” movement, where more and more companies are saying, well, basically, if I have to support smartphones and tablets anyway, and my CFO is probably carrying around an iPad and all my new employees are coming in with iPhones, so I have already got to support this stuff, so then I might as well encourage it. And I might as well basically have a model where instead of issuing a company laptop to everybody or even a company phone, why don’t I just let people bring in whatever device they want and just plug-in and access it. And then they get all excited, because then they say, well, not only are my employees going to be happier and more productive, but then I don’t have to buy them hardware anymore, so I can cut my budget. So that’s the big thing that’s starting to happen right now.

Alexia Tsotsis: So how does it affect the way people are building? I have about five companies that have made this list. Some of them are yours — Okta, which is big on the Bring Your Own Device thing, because you are logging in through Okta. Okta, Cloudera, Box, GitHub, Zendesk, and Asana. Are there any that I have missed?

Marc Andreessen: We just invested in this company called ItsOn, that we announced yesterday.

Alexia Tsotsis: Oh, the mobile –

Marc Andreessen: The mobile billing. The advantage — the thing that that’s going to be able to do is do split billing in a new way, between the business and the consumer. So on a single device you will be able to cleanly build data usage by application. So your employer can pay for your Salesforce.com and your Workday data usage on your phone and you get to pay for your Facebook and your Hulu usage. So that will be another enabler for a lot more of the Bring Your Own Device stuff.

Marc Andreessen: We have a bunch of stealth investments. I mean, this is a big, big thing — big change for us, so we have a bunch of stealth investments — I mean, companies that haven’t talked about what they are doing yet. Who else? I mean, there is a bunch, who else should we have –

Alexia Tsotsis: Plattfora.

Marc Andreessen: What’s that, Plattfora? Yeah, Plattfora is the actual user interface layer on top of Hadoop. So sort of Plattfora and Cloudera kind of go hand in hand. Actually — we have another one actually that is — well, it sounds esoteric, but it actually is very relevant. We have a company called Tidemark, which is in a category. It’s called Enterprise Performance Management, which is kind of a weird term. It’s basically large-scale financial planning and analysis for big companies. The significance is it has a — I believe it only — I don’t know if it only or primarily, but I think it only has an iPad UI. So it’s the first complex financial system for big companies, where the assumption is that the user is on an iPad. That’s a really big deal, because that category of software, line managers and businesses have never actually used that software themselves. Instead they employ analysts to use the software who become highly trained on the software. By putting the iPad UI all of a sudden you can have anybody in the business have access to all the financial analysis and planning. Even in a very deep sort of sector of enterprise software where most people would never see it, this change is having a big impact. What else? Asana you mentioned.

Alexia Tsotsis: Asana, Box, Zendesk, these are the companies that I am assuming I will be using four years from now to run my business.

Marc Andreessen: Yeah, exactly! Exactly right! Then of course Workday, of course, Salesforce.com, of course NetSuite, 37signals. We probably have three or four others in our portfolio that I am blanking on, but yeah, this is sort of the — and then by the way, the corresponding thing is that a lot of this is on how you run a business and then how you do marketing, of course; AdWords and Facebook and Twitter, all these systems and then all the enabling systems for that; so HootSuite and Marketo and –

Alexia Tsotsis: GoodData.

Marc Andreessen: Oh, another one, GoodData. So GoodData is at the intersection of kind of marketing and business. So GoodData is an actual easy-to-use analytics package. It’s sort of like a supercharged version of Excel that lets you suck in data, you can suck in all your Facebook advertising campaigns, you can suck in all your Salesforce.com data, and you can run — you can actually, yourself, as a small business person, actually analyze and find friends and data.

Alexia Tsotsis: I have heard good things about them and they just sent us a guest post.

Marc Andreessen: They are very good. So then you add up all these companies and you are like, “Well, okay, so number one, they are all basically new companies. I think who is not on that list are all the existing companies that sell business software.”

Alexia Tsotsis: SAP, Oracle … I mean I wrote a post about this that was supremely misunderstood and then today SAP came out with SAP Jam, which is a competitor to Yammer and to Salesforce, but it’s their own socialized CRM, like HR management software. I worry about this because it’s not going to work. You can’t fight the future.
Marc Andreessen: Oh, right, right, right. I mean, the joke about SAP has always been, it’s making 50s German manufacturing methodology, implemented in 1960s software technology, delivered to 1970-style manufacturing organizations, like it’s really — yeah, the incumbency — they are still the lingering hangover from the dot-com crash. So a lot of incumbent business software companies did what a lot of big companies actually did and other industries, media companies after the dot-com crash, which is they said, “Oh, thank God we don’t have to worry about this Internet thing. It’s over. Stick a fork in it. It’s not going to be a big deal.” And then it turned out that it actually wasn’t over, and they still haven’t adjusted.

Alexia Tsotsis: Yes, and we are watching that now. And so the other reason that I am very interested in delving deep into this space is that it seems like IPOs like Workday, Palo Alto Networks are sort of — they have metrics and analytics that Wall Street understands, more so than a Facebook; like “We are going to sell X number of this in the next year.” So it would seem like they are an antidote to, or at least less offensive than, social/consumer Internet companies are to the public markets.

Marc Andreessen: For now. The whole market goes back and forth in whether they prefer enterprise businesses or consumer businesses. The argument in favor of consumer businesses is you don’t have these crazy end of quarters like when the IT purchasing manager doesn’t buy the product and the company misses the whole quarter. The advantage of the consumer businesses is they tend to be much broader-based, much larger number of customers, that tend to over time be a lot more predictable. The advantage of the enterprise companies is they are not as subject to consumer trend, fad, behavior. But I would say the market is schizophrenic. So right now we are in an era where the market wants enterprise companies. I am just saying like wait a year, that will flip again; wait another year after that, that will flip again. It’s sort of the picks and shovels thing. Like everybody — it’s like the consumer businesses get really hot and then everybody realizes that there is lots of competition and that those models have — they are complicated businesses and they have their issues, and then everybody gets all excited about picks and shovels. And everybody rediscovers the picks and shovels analogy and says, “Oh, the Gold Rush in California, the people that made all the money were the guys who were selling picks and shovels to the prospectors.” And then people realize the picks and shovels business is really hard, and then everybody says, “Oh, we should invest in the consumer company because they” — so it’s just — Alexia Tsotsis: It’s cyclical.

Marc Andreessen: It’s cyclical. It’s deeply cyclical. But we are in an environment right now, to your point, where there has been huge rotation out of the consumer companies into the enterprise companies.

Alexia Tsotsis: It seems like the consumer market is starting to cool — I mean, not starting, but the signaling is there.

Marc Andreessen: Yeah. It’s unpredictable. All you need is for one of the new enterprise companies to completely whiff a quarter and their stock will collapse and then everybody will get all freaked out. I mean, it’s just a continuous — the reality is every single business is hard.

Alexia Tsotsis: I love this.

Marc Andreessen: There are no easy businesses in the world other than maybe Google, but other than that, there is no easy business anywhere in the world. So what happens is Wall Street gets enamored by the businesses that look like they are easy, until it turns out that they are not, and then Wall Street gets disillusioned and freaked out, and then rotates into the businesses that they think are going to be easy, and then they get endless disappointment. It’s like a seventh or eighth marriage at some point. At some point the problem isn’t with your seventh wife. At some point the problem is with you.

Alexia Tsotsis: Is the solution “keep calm and carry on,” or is what the solution to this?

Marc Andreessen: There is no solution; it’s a permanent state of affairs. So this is a big part of what actually we do. A big part of why venture capital actually is important and enduring is because the public market is flighty and late-stage investors are flighty, and customers for that matter are flighty, and so you can’t — if you are running one of these companies you can’t — you just can’t rely on people being balanced. They are just not going to be. And so you have to have a level of determination to just stick through the good times and the bad times. And you need to have investors at the core of your company who are going to support you through that. The big advantage that we have as a venture capital firm over a hedge fund or a mutual fund is we have a 13-year lockup on our money. And so enterprise can go in and out of fashion four different times, and we can go and invest in one of these companies, and it’s okay, because we can stay the course. And then what happens is everything tends to get better, all the products tend to get better, all the companies tend to get better over time if they are working hard at it. So we are fine. Like if everything we are investing in goes out of fashion, we are not going to change anything we do, because we can’t change anything. We are already invested in these companies; we can’t sell our stock. We don’t have to sell our stock. So we just say, we will go back to work. And then at some point it really gets exciting again.

Alexia Tsotsis: I guess the trick is to be hyper-aware.
Marc Andreessen: So the big thing we try to do is be aware of the difference between the reality and the psychology, and the reality tends to progress in a certain way and then psychology tends to whip all over the place. It was very educational for a lot of us to go through the dot-com crash, because you remember, in 2002, like there were a number of universal truths asserted in 2002; the Internet didn't matter, consumer Internet business was dead. Larry Ellison in 2002 came out and gave a speech and said the correct model for enterprise software, enterprise computing, will last for 1,000 years. He said all these kids that were trying all this new stuff and it didn’t work, and now we know it didn’t work, and so the model is going to be the existing IBM and Oracle for the next 1,000 years. And everybody kind of said, hmm, you know, that makes a lot of sense, like all that innovation stuff didn’t work, and so —

Alexia Tsotsis: That's what David Sacks said.

Marc Andreessen: Exactly, this is the fact. People reach a point where they start to get a little bit too rich, maybe a little bit too old, and they start to say these things. And then so here we sit 10 years later and we are in the middle of a complete reinvention of everything in enterprise computing, and it’s like, okay, like that’s the reality. People happen to be excited about it again at the moment. That's great. I am happy for that. But wait two years and they will be depressed about it again, but that won’t keep it from happening. It will still happen.

Alexia Tsotsis: It's just like the fashion industry. So because it's heavily fashionable now, do you see it being over in a year?

Marc Andreessen: No, I don’t mean to make a specific prediction. I don’t know if it’s a year, two years, four years. Look, all of the products are going to keep getting better. All of the trends that we are talking about are going to keep continuing. Nothing is going to stop consumerization of the enterprise. Nothing is going to stop Bring Your Own Device. Nothing is going to stop Software-as-a-Service. Nothing is going to stop cloud. All those things are just going to keep going. I am just saying people are going to be — they are all excited about them now. At some point again they will be unexcited about them and then at some point after that they will be excited about them again. So it’s hard to draw conclusions about the importance of the trends or the progress of the trends by the current level of press coverage, the current level of Wall Street enthusiasm.

Alexia Tsotsis: So beyond the press coverage, beyond the fickleness of trends, beyond the application layers — because most of those companies are just apps — what are the real opportunities you see in the enterprise stack as it stands right now?

Marc Andreessen: Well, there is a whole bunch. So there is a big thing — there are a couple of big things that are happening. So one of the really big things that’s happening is, historically the best enterprise technology was only — it’s a trickle-down thing — the best business technology was only ever available to the biggest companies. And so if you were a Fortune 500 company with a big IT department, you had a huge advantage over a small business that was trying to compete with you, because you just had so much more budget and staff and professionals and expertise and access to all these big vendors and you could spend tens and millions of dollars on all this stuff. So it was very easy for — in the old world it was very easy for big companies to use IT as a weapon against small companies. The classic was Walmart versus local retailer, right? Walmart’s advantage in logistics and in pricing and in data analytics was just so great that they could kill small retailers at will. Today all the consumerized enterprise stuff is as easily usable by the small business as it is by the large business. In fact, it’s probably more easily usable by the small business than it is by the large business, because with a small business it's like you can just use it, like you don’t have to go through a long process, you don’t have to have a lot of meetings, you don’t have to have committees, you don’t have to have all this stuff, you can just start picking up and using it.

So the best technology for inventory management and for financial planning and for sales-force management and for online marketing can now be used just as easily or more easily by a small business. There is an opportunity here for a shift of the balance of power for big businesses to small businesses. And then for vendors, the companies we fund, there’s an opportunity to really dramatically expand the market, because a company like Oracle, as successful as it is, it only really has about 5,000 customers that really matter worldwide. Whereas, a company like Box or a company like GitHub could have 500,000 customers or 5 million customers that really matter, and that’s a huge change.

So market expansion, small business versus big business, what else? Oh, the shift, the other big one, the shift from CAPEX to OPEX. So the shift from buying a lot of servers and databases and software licenses and networking equipment, the shift instead to just renting it all. So the shift towards cloud services. So we don’t have — no company that we invest in anymore actually ever buys any hardware. I mean, they buy their laptops and that’s basically it. And increasingly they might not buy their laptops, because their employees will just bring their own devices. But they don’t buy servers. They don’t buy storage devices. They don’t buy any of this stuff, they just rent on AWS. And they don’t buy sales-force automation software, they rent on Salesforce.com.

And so having sort of a much lighter-touch way for businesses to be able to get funded, you just need a much smaller budget. And that’s why you see these — you see it in the startup world, you see three or four kids with laptops.
Marc Andreessen: All the above. They are all changing. I think they are all changing.

Alexia Tsotsis: What do you think about the interplay between the enterprise market becoming more efficient and the explosion of the consumer market because you don't have to pay for something like storage?

Marc Andreessen: I don't know, it's sort of all intertwined. I mean it's all — because a lot of what businesses do is then offer consumer services based on all these changes. So it's kind of all — that's why I say it's kind of all happening at the same time, a lot of the same stuff.

I would say the consumer Internet companies — in a lot of ways if you go inside the consumer Internet companies and you see how they run, it's how all their businesses are going to run. They are going to be doing all of the same kinds of things. The big businesses are just in the process of trying to figure out how to catch up.

So everything, Hadoop and scale-out architectures and cloud services, and the whole thing it's all — and use of new technologies like Box and GitHub, the consumer Internet companies all are just built this way. And then if you go inside a big consumer product's company or a big manufacturing company, they are all trying to figure out how to make the jump. But it's all kind of the same stuff.

Alexia Tsotsis: So which of the big incumbents do you think are most likely to get disrupted by this new wave of the enterprise cool kids?

Marc Andreessen: Yeah, this is the part where I get into the most trouble.

Alexia Tsotsis: That's why we save it for the end.

Marc Andreessen: Yeah, exactly! I don't know if I am going to — let's see, I am going to try and figure out if I am even going to answer the question. So I would say for sure — like the systems companies, like the companies that provide hardware, the server companies and networking companies, the bad news for them is the end customers are not going to buy as much stuff; the good news is the cloud companies are buying a lot of stuff.

So for every server that's not bought at a manufacturing company, there's a server being bought at Amazon. So it's a change in purchasing pattern for all the gear, but the gear is still being bought.

I think it's at the software layer where the big disruption happens. I think it's application software in particular and just sort of an extended infrastructure software. It's like anything for which there is a — any piece of installed software for which there's a web or a cloud equivalent, I think is in real trouble, and I think that's just now becoming clear.

The other thing that's happened is 2012 seems to be the year of the actual SaaS tipping point, like where big companies are now saying, you know what, it's fine, like I can do it, I can do Salesforce, I can do Workday. Because there used to be lots of issues around can I trust the security issues or liability issues, and an awful lot of big companies are now saying, “You know what, I am going to save so much money, the service is going to be so much better, my users are going to be so much happier, more productive. I have got to make this stuff work on iPhones anyway, so I have got to do something new.”

“My old software vendors are charging me these huge upgrade and maintenance prices. I can switch to SaaS for less than the cost of the maintenance on the old software.” Like, at a certain point it becomes —“Oh,” and then on security it's like, “Yeah, I may have concerns about SaaS security, but it turns out I have the concerns about my own internal security anyway.”

So every one of these companies has had an employee steal a laptop that has 25 million customer records on it, and they are like, “Well, okay, if I can't even lock that down, then why am I that worried about whether somebody is going to break into Salesforce.com?” And by the way, Salesforce.com has gotten much better at security. So there is a bunch of new technologies coming out that are going to make cloud and SaaS even more secure, and I think are going to end up making — I think cloud and SaaS are going to end up being a lot more secure than anything inside the firewall. So that's the other thing that's about to happen.

Alexia Tsotsis: So which enterprise companies are doing their best to adapt to just this tidal wave of trends and which ones are just completely failing?

Marc Andreessen: The problem is I have conflicts on this issue, because I am on the HP board in particular, so I can't really — unfortunately I am kind of gagged on the topic of the big companies.

Alexia Tsotsis: Are you happy with how HP is doing?

Marc Andreessen: This is exactly what I can't talk about. I just can't talk about it. So the problem is I can't talk about HP and I can't talk about HP's competitors, so it's just a no-fly zone for me.

Alexia Tsotsis: I respect that.

Marc Andreessen: So I have to stick to the startups.

Alexia Tsotsis: Let's see, what about other companies that aren't in your portfolio?
Marc Andreessen: Although I have a lot of opinions. You mean startups?

Alexia Tsotsis: What about startups that aren't in your portfolio, because you said that only 10, 15 companies a year are responsible for 97 percent of the returns. Which enterprise companies that aren't in your portfolio are you interested in?

Marc Andreessen: So let's see, there is this category of kind of outsourced work.

Alexia Tsotsis: TaskRabbit.

Marc Andreessen: Well, there's TaskRabbit and Zaarly and companies like that on the consumer side; and then on the business side there's eLance and oDesk and RentACoder. So these companies that are kind of for — in the sort of mechanical term, distributed workforces and outsource work being run online. So like oDesk, oDesk you can actually have remote contractors working on a project, and one of the features is that it actually takes snapshots of their screen every five minutes. You can see if — anybody who actually manages anybody, number one that sounds spooky, but number two, “Wow, that sounds great, like, I sure wish I can do that.”

So there is sort of the whole category of an outsourced workforce that sort of — it goes back to what you said about the employees is, you will have — it feels a lot like in the new economy you will have a lot more contractors. You will have a lot more people with sort of fluid careers contracting on a project basis, and then all this technology is going to be an enabling layer for that.

So anybody on their laptop, anywhere in the whole world, being able to tap in and be able to get work and do work, whether it's for small companies or big companies like that. There is a whole layer of software there. We haven't seen anybody really punch through on that yet, but I am very fascinated by it. We haven't made an investment there yet. That's one layer.

Let me think, what else haven't we done? I mean, Cloudera is I think a good — we haven't actually done an investment at that. We haven't done an investment at the Hadoop layer. We have done — Platform is our investment, which is the intelligence layer above Hadoop, but Cloudera definitely deserves to be on the list.

Zendesk and kind of its generation of companies are definitely for real, or so it appears.

What else? We have been pretty active. I mean, we have been trying to take down mostly good companies. We haven't done anything yet with this whole category of marketing, the new marketing software so like Marketo and HootSuite and companies like that, we haven't really done anything yet, but that's a big deal.

It's sort of like — if you are starting a new company it's so obvious that you would want to do most of your marketing on Google and Facebook and Twitter, whereas a lot of the existing companies still haven't wrapped their heads around that.

Education — there is actually going to be more and more. So actually companies are going to get a lot more interested in education for two reasons.

Number one is, a lot of companies need to actually educate their customers or their partners, and a lot of that has to happen online.

And then the other thing is companies are having — if you talk to anybody running a company, they are having real trouble hiring enough qualified people. So companies are going to have to take a more direct role in educating the candidates or educating their current employees.

So the sort of model of employees just show up and they are either educated or they are not is not working very well. There's lots of mismatches. It's one of the reasons unemployment is running as high as it is, is people just don't have the skills they need for the jobs.

So I think employers are going to have to get a lot more actively involved in making sure that the supply of candidates is actually educated and that they can hire somebody who doesn't yet know what they need to know and actually educate and train them, and a lot of that is going to happen with the new technology.

So we have this company Udacity as an example, that's going to be, I think, important in all of that.

Alexia Tsotsis: I think the model there is if someone shows up and they have got 80 percent of the skills.

Marc Andreessen: Yeah, let's teach them — right, exactly, the employer says let's teach them the other 20 percent. And it's like, well, instead of literally sending them to college, which presumably didn't work the first time around or whatever, let's just go ahead and provide them with the online training. Let's set them up with their tablet at home with high-definition video. They can develop their remaining skills, or be able to retrain people once they are in the jobs.

The other is there is this real issue, like for some people it feels great to never be tied to a specific employer and to always be doing contract work and be changing jobs every two years, and it feels like it's fun and exciting and exhilarating. For a lot of people that's really scary. And so the lifetime employment promise that the big companies used to be able to make was very compelling for a lot of people because it felt safe.

So now you are in a world where the big companies can't deliver — even if they wanted to deliver on lifetime employment, they can't, and so then they have got sort of two choices.

One is, do they start to basically be a lot rougher with their — they start to do a lot more layoffs, a lot more restructurings. I remember IBM — I don't think IBM had a layoff for 50 years. And I was actually at IBM — I was an intern at IBM when they were ramping up for their first layoff I think they had ever done, and, like, the level of freak out in the company was beyond belief. And people had no idea what to do if they got laid off from IBM. And it turns out their skills weren't actually very useful to work for any other company, because IBM was so unique in how it ran.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
So I think the companies have a real question about how do they develop their workforces, how do they make sure that their employees stay relevant for the purpose of staying inside a company for a longer period of time? And then how do you get the workforce over time to be a lot more flexible and adaptable, so that if you have to layoff a ton of people, or if you have to get out of a line of business, or if you have to expand into a new business, how do you get your current employees to adapt better to that?

Alexia Tsotsis: Do you think that's the most ripe-for-disruption area in the enterprise currently?

Marc Andreessen: I don’t know if it’s the most, but it’s a big issue for every company. It’s a big issue for companies, because companies have hundreds and thousands of employees, it’s like, yeah.

Alexia Tsotsis: What are the top three issues that startups don't exist for yet, because that sounds like one that a startup doesn’t exist for...

Marc Andreessen: Sort of. Education is a big component of it, yeah, it’s possible, it’s possible. I don’t know. We will wait for the entrepreneurs to answer that question.

Alexia Tsotsis: Probably the biggest enterprise cool kid is GitHub?

Marc Andreessen: They are a big one, yeah.

Alexia Tsotsis: And you made a major investment?

Marc Andreessen: Yeah. We think it’s the largest investment ever done.

Alexia Tsotsis: How did you convince them to take your money?

Marc Andreessen: That’s the key thing. So they were beating off venture capitalists with a stick. So they actually — I don’t know if you remember this, they used to have to put on their website, they used to say — they had four metrics that they would put on their website. They had, I think, it was number of users, number of projects, number of code check-ins, and amount of venture capital that they had raised, and that final number was always zero, and they were really proud of that.

The GitHub guys did an amazing job. It’s very rare actually to find a main — it’s very rare to find an important company that never raised any money. It’s very rare that they actually successfully bootstrapped, because it’s just so hard to do if you can’t invest any money.

So what they did was incredibly impressive. They reached a point though where they decided that they had the opportunity to become a very big and important company. And again, I would say there was a top-down and a bottom-up reason for that.

The top-down reason was they are the place, we think, and they believe, they are the place where all the software code wants to live. They are the place where all the open source code increasingly lives. All other code increasingly uses a ton of open source code. And so all the software basically wants to be in the same place, and it wants to be in the place where all the open source software is.

So they have an opportunity to be the main company that provides the systems for developing software, number one, which is just a very big opportunity, and they really decided to go for it, and that requires investment on their part.

And then the bottom-up reason was because they have enterprise customers lining up, like they have enterprise customers bombarding them with interest in buying services on GitHub. And they did not have — at the time we invested they didn’t yet have any sort of sales or marketing kind of motion to be able to do it on. They didn’t have a Salesforce, they didn’t have the sort of pricing plans — the whole thing to be able to do that — and we have a lot of experience with that.

Alexia Tsotsis: So who are the enterprise companies and do they have...

Marc Andreessen: Tons, it’s like the who’s who. I mean, they gave us access. One of the things they did in the diligence process was they gave us access to the email box that had all the incoming messages from all the CIOs and purchasing managers and all these big companies. And it’s literally like, “Hi! I am from big bank X and we already have like 600 people on GitHub and we want to buy an enterprise license. Who do we call and where do we send the check?” And they just had the email queue up this, and they didn’t have — they weren’t — like it’s just sitting there.

And so we are working with them to help them build out the sales and marketing capability to be able to really go get all that business.

Alexia Tsotsis: And that’s how it will make a billion dollars a year?

Marc Andreessen: Yeah, yeah. Well, I don’t know, we will see how. I mean, aspirationally, yes. It should be a very big business. Historically companies in that market have been very successful. Big one, Rational is a big company that IBM bought a while back that’s in the same market.

And then even companies, Mercury Interactive was a big company that HP bought that was in a similar kind of market. And then in the old days you had companies like Borland and Lotus that were very big in these markets. So this is sort of the new version of all that. So if it works it should be very — I mean, it’s working, so it should be very big.

Alexia Tsotsis: So we are watching the collapse of Zynga and Groupon and LivingSocial in the consumer space. Where is the bloodbath going to be, if there is one, in the enterprise space?
Marc Andreessen: I don’t know. Probably in the second and third tier. I mean, usually when you have a funding boom, categories usually get overfunded. So probably it’s in the second- and third-tier competitors. We actually get yelled at for this a lot, but we really believe it. So the big technology markets actually tend to be winner take all. There is this presumption — in normal markets you can have Pepsi and Coke. In technology markets in the long run you tend to only have one, or rather the number one company in — the number one company in any consumer products — cars, the number one company in cars is, I don’t know, Toyota or whoever it is.

Alexia Tsotsis: I think it’s Toyota.

Marc Andreessen: Fifteen percent or 18 percent market share. The number one soft drink has only 60 percent versus Pepsi, but like what is Coke as a percentage of all drinks, it’s, I don’t know, maybe 10 percent. The big companies, though, in technology tend to have 90 percent market share. So we think that generally these are winner-take-all markets. Generally, number one is going to get like 90 percent of the profits. Number two is going to get like 10 percent of the profits, and numbers three through 10 are going to get nothing. And the problem is, of course, there is too much venture capital, and so companies three through 10 still get funded. So there are probably lots of sort of second- and third-tier companies that are getting funded right now that won’t succeed, but that won’t have anything to do with whether or not the winner succeeds.

Alexia Tsotsis: Which is all the venture capitalists —

Marc Andreessen: Well, the venture capitalists who are successful in investing in the winners will be very happy with this. The venture capitalists who are investing in the losers will be very sad. But everybody will get freed, because at some point there will be a bunch of companies — a bunch of startups that will go bankrupt, and then everybody will say, that must mean the whole sector is going down. I just think people get confused. People get confused about — it’s really funny watching the stock. Like the stock market does this all the time. It’s like one Internet advertising company will have a bad quarter and the other Internet advertising companies’ stocks will all drop.

Alexia Tsotsis: It’s tethered.

Marc Andreessen: Maybe, but maybe it’s because the other ones are now taking market share away from the one that had a bad quarter. So I find the markets all have trouble processing cause and effect in the short-term and people get all confused.

Alexia Tsotsis: What is the solution to that? It’s so perplexing.

Marc Andreessen: That’s permanent. I think it’s permanent. I think it’s human nature. There are certain things that can’t be fixed, and I think that’s one of them.

Alexia Tsotsis: One would argue that the enterprise has come back into fashion, because the market is cooling on the Instagram deal being a billion dollars and now being $700 million. The market is cooling on all these photo-sharing apps and just ways to update your friends about your whereabouts. Is the only reason the enterprise is sexy because it actually makes money and has results, or is there some overlaying tide of innovation that’s happening that’s exciting people simply because it’s culture-changing?

Marc Andreessen: I think it’s all the above. I think the changes are real. The businesses are good, which is nice. And then I think it’s also sector rotation. We talk to a lot of the big hedge funds, mutual funds. It’s really funny. We are talking about big hedge funds, mutual funds, about six months ago they all started saying, well, you know, we really think there is going to be a rotation from consumer and enterprise, and we are going to really get ahead of that. And I am like, yeah, you and 10 other guys in the last two weeks have told me the same thing. It’s like, good job, you are way out ahead on the leading edge on this. I get to the point where I am just like — my running joke has been, it’s like little kids, like everybody out of the consumer pool, everybody into the enterprise pool. So everybody out of the waiting pool, everybody into the hot tub. What happens when you have capital flowing in a rotation is, if all the capital starts to leave one sector and go to another sector, then all the stock prices rise in the sector that all the capital is going into, because everybody is buying those stocks. And then everybody says, wow, look at how much those stocks are going up. We should invest in those stocks. And so the up cycle starts perpetuating. And the down cycle similarly — a lot of reasons people have been selling out of the consumer stocks is because they have been going down, because people have been selling out. So the cycle is perpetuating.

Alexia Tsotsis: So what should the smartest entrepreneur do? You have $1.2 billion to spend. Where are you spending it?

Marc Andreessen: Investing. So the smartest entrepreneurs I think generally ignore all this. We really look for the entrepreneurs who don’t pay any attention to this. We really look for the entrepreneurs who say the following, they say: “I have this really good idea and I know it’s a good idea for the following eight reasons, and I have thought about it and I have worked in the field, and I know what I am doing, and I have talked to the customers and I have figured it out, and I am going to do it. I am just going to flat-out do it. And I am going to do it whether you fund me or whether you don’t fund me or I don’t get funded. I am still going to do it.” That’s the entrepreneur we are looking for.
Sometimes that entrepreneur is in a sector that's completely dead, and that entrepreneur is going to say, I know that everybody thinks that this sector is just dead, and in fact that's probably why now is a good time for me to do this — is because I am not going to have very much competition. Well, so a lot of these companies you talk about now, like Workday and GitHub and Box, got formed and funded when everybody thought enterprise was dead. Just like the consumer companies like Facebook and Twitter got funded and everybody thought the consumer stuff was dead. So sometimes you get the entrepreneurs who are actually counter-cyclical. They are doing it precisely because nothing else is happening in that sector, and that means the big opportunities are just wide open.

On the other hand, sometimes you get the entrepreneurs who say, "I know I am doing the 36th workforce collaboration system in the consumer side; Google was the 36th search engine, and I know it's 1999, and I know the whole sector is overfunded, but I have a better idea. I know how to do it better, I know how to do it different. I have learned from the mistakes that everybody else is making, and I am going to be the winner."

And so we will fund either one. We fund companies in the hot sectors and we fund companies in the not-hot sectors. The only difference is the pricing, and the pricing varies basically by like 4x. But what we have just found or what we have sort of tried to learn from history is you can't — for what we do you can't really time the stuff. And the last thing you want to do is look at what's happening in the public market today. And this is the thing that's weird about how venture capital works — a lot of venture capital investments are decided based on whatever the NASDAQ is doing.

Alexia Tsotsis: Are you seeing down rounds because the NASDAQ is down?

Marc Andreessen: No, we have not seen down rounds yet. We have not seen down rounds yet, but consumer rounds that are happening now — consumer growth rounds are happening now at 2-4x lower prices than they were six months ago. And the enterprise pricing is up a lot. Enterprise pricing a year ago was probably half what it is today. Maybe a third in some cases, and that made us happy at the time, but it's all good.

Alexia Tsotsis: Last question — your top five enterprise cool kids?

Marc Andreessen: I think they are all on your list.

Alexia Tsotsis: They are on my list?

Marc Andreessen: Yeah, yeah.
13. Being a great founder
Lecture 13: How To Be A Great Founder

http://startupclass.samaltman.com/courses/lec13/

Reid Hoffman

When I look through the syllabus of this class and thought what would be useful skills, what I’ve been thinking about is how do you see yourself as a founder? How do you think the skill set is? And what are the things you should be thinking about in terms of: am I ready? How do I get ready? Is it the right thing for me? These sorts of things.

So let’s start with the perception of what a great founder is. And classically this tends to be Steve Jobs, Bill Gates, Elon Musk, Mark Zuckerberg, Jeff Bezos. And it’s an image of founder as Superwoman, or Superman, who has this panopticon of skills. I can use the word panopticon because I am here in Stanford. It’s things like, I can do product market fit. I am great at product, I am great at strategy, I am great at management. I can fundraise. I can do all of these skills and the thing you are looking for in a great founder, in the idea of the founder as a super person, is I am looking for someone who is awesome at all these things. They are well rounded, they are diverse. They can bat on all skills.

And part of how I found this emphasized, in the beginning of my own entrepreneurial journey, I remember reading an article that said Bill Gates is smarter than Einstein. And you are like, Bill Gates is really smart and is very accomplished, but I am not sure smarter than Einstein is really a phrase that even Bill Gates wants to be next to. It’s partially because of this image of a founder as super person. Which is, a great founder can do anything. Jump over tall buildings in a single bound, all of these sort of things. And the reality is that a founder is someone who deals with all these different headaches and no one is universally powered.

Generally speaking you hope to have a couple super powers. Some things that are a unique edge to you, some things that are unique to the problem you are trying to solve, some things that will help you get an edge. Because competitive differentiation and competitive edge is super important. But it’s not actually a function of genius.

And frequently it’s very hard to tell the difference between madness and genius because usually it’s the results that play out. Sometimes when dealing with uncertain environments you may even be genius and later may be thought to be a mad person. Or you may be a mad person and turn out lucky, you may turn out genius. It’s actually a challenging set of how do you think about these sets of skills. And when us, mere mortals come into this sort of battle, what is the right way to think about it?

So when I thought about this question, on how is one a great founder, these are all skills that are super important. These are all things that you say, okay, this is really really important to do and you must in fact do this well. And it begins to look like a superhuman task. And what I did was decide to take a superset of these and focus on the interesting things to think about. What is it that actually makes a great founder? Because it’s not actually that you score ten out of ten of these, you become the entrepreneurial olympian. You are actually the best at all these things.

So lets start with team. One way, I think, to explode the myth of super founder is usually its best to have two or three people on a team rather than a solo founder. Its not to say that solo founders don't actually play out and they can successfully. But most often two or three people is much better. When I look at these things as an investor, and I say what is a good composition of a project and founders that are likely to succeed. Its usually two or three of them. And the reasons are, we have already talked
about that there's this very broad set of skills. There is this whole set of questions on how you adapt your company or be successful. If you have two or three founders, you have different skills you can compensate. Because by the way everyone has weaknesses. You can compensate for each others weaknesses. In the diversity of problems you encounter as a founder, you can actually attack them.

Other things I suggest when you look at, essentially, a founding team, is to have a real high preference for having co-founders, having a high degree of trust for those co-founders because one other way on the whole entrepreneurial thing to die is you get a year down the road with your co-founders then you are going through a messy divorce. That is not always but frequently fatal. And also the diversity of the tasks that you do. The next thing is location.

Frequently, I have been told to me. Oh Silicone Valley aggregates all this super talent, which it does. The reason why Silicon Valley startups are so successful is because all of these great people--immigration which is hugely important for talent and founders that immigrate here. Now if you think about it from basic math, even if you take something that Silicon Valley is super strong at, which is essentially software skills in the last two decades, not all the great software company move here. Not all of them can move here. There are many of them in other various parts of the world. And so why do I put choice of location up there? its one of things it comes down to in thinking whether or not you are a great founder. Well the reason is, because of what great founders do is seek the networks that will be essential to their task. And they realize its not just about, I am superman, I can do this anywhere. I can do this in Antarctic, etc--In order to be successful, I have to go to where the strongest networks are for the particular kind of thing that I am doing. And Silicon Valley, by the way is super good at some kind of tasks, some places that you essentially try to solve certain types of problems. But its not good at all of them. Let me take two examples.

One is Groupon, I don't think Groupon could have ever been founded here. Even though its a software product, it even generates a network. Obviously a lot of the great networks are here. And uses internet technology as a mobile product and everything else. All of which we have a lot of great skill here in Silicon Valley and the networks are really good for this. One of the things thats central for Groupon for its early days, was having massive sales forces. And massive sales forces strengthens and weakens of a workforce tends to go together. Silicone Valley, tends to be pretty adverse to plans that involve, Oh we are going to rent a twenty-five story building and in twenty of those stories we are going to have floors of sales people. And thats how we are going to get our thing going. That kind of plan here tends to not get a lot of interest, tends to get a lot of criticism, tends to not have tautography to it, tends to have financiers talk about things like cap efficiency, and network effects, and other kinds of things that are key here. And so its actually not a surprise that actually in fact, Groupon was required to be in Chicago which is really good at this, as a way of getting going and showing that even software start ups can be in other places. But even if you begin to think about it, you say what kind of other kinds of startups would someone be an idiot to move here to do.

Think of someone doing a fashion start up, not fashion a la poshmark which is a mobile marketplace that are a bunch of things good here. But I'm designing a new fashion company. And I'm going to come to Silicon Valley to do it. Thats actually not sure a great idea. The fashion company might be a great idea but you want the networks that support what you are doing. So part of the reasons why, where should I locate my start up, is a test for thinking about am I great founder is because part of the thing that happens when you are founding a company is--I will go to where this is successful to do. The metaphor I often use for entrepreneurship is jumping off a cliff while assembling an airplane on the way down. And the reason I do is because its hard, it has a quasi mortal exit, which are default dead so you start taking every possible chance to actually win. So great founders will move to where the network is. This network is graphically Silicon Valley for tech startups, for mobile, for marketplaces this is a really good place to do it. For a bunch of other things, you should think about a different location.
Now here is something that's very in vogue. Very conventional to say you're a contrarian these days. So let's talk a little about what what contrarians actually is. So it's actually pretty easy to become contrarian. Its hard to be contrarian and right. Particular, when you are thinking about is my idea is contrarian or contrarian enough. How does a smart person actually disagree with me? Because if you can't think of a smart person who isn't just ignorant or just crazy or anything else. But is a smart person that is somewhat expert may think that your idea has some serious challenges than it actually isn't contrarian. Contrarian is relevant to an audience. So when you are thinking about contrarian in terms of a really good contrarian idea is like, say its consumer internet, okay what would other consumer internet people think is actually not a good idea. And part of when you think about contrarian is to say, okay what do I know that others don't know? Because it isn't just, oh I'm brilliant and other people aren't its the reason my contrarian thing is right. Its a very bad test, that happened to be true of course lightening can strike you in the field. So think a lot about what is it that I know that other people don't know. For example, in the very early days of Linkedin, part of what I advise all founders to do was talk to every smart person who will talk to you and give you feedback. So with Linkedin I walked around and said here’s my idea, what do you think?

Two thirds or more of my network, including some of very smart people, all thought I was nuts. The reason why they thought I was nuts was because I said its a network product, its only valuable with a bunch of people in it. The first person has no value until they invite the second one. Second person, first person no value in it they already know each other. When do you actually begin to deliver on your use case which is 500K to a million people. And so you are never going to get to size. Its never going to grow. Now what I knew was that the critics didn't know was that I could think of a set of different way by which people would say hey look I believe in the vision of this. I think its interesting, or I think a product like this should exist, or I'm willing to play around with it. And I can level those sets of interest to grow the network to get to enough size that you can begin to deliver on the value propositions in which Linkedin had. And that was the specific thing that I knew that the critics weren't thinking about. So when you think about being contrarian, you have to think about how is it that smart people disagree with me that disagree with me from a position of intelligence. And there is something that I know that they don't know that will actually play out to be true. Now in this case, in general, as a founder its good to be contrarian in the real sense.

Now the last part on the contrariness is to think about there are lots of different ways to be contrarian. For example, a frequent one will be, oh yeah that’s a good small idea but actually that’s not small its large. Or actually in fact you can assemble the talent or while most consumer cellular start ups tend to be, another LinkedIn example, only successful with the rocketships, actual a compounding curve can be very very valuable. LinkedIn never had its rocketship moment, it was compound year by year. But in consumer internet that becomes Atypical in the pattern.

So here you begin to get to a bunch of sorts of problems that essentially founders run into. well should I be doing the work? Or should I be recruiting people and delegating the work? And classically the answer to this is, actually in fact you need to do both. In fact, not only do you need to do both, you need to sometimes do one at 100% and sometimes the other at 100% and even though this is not so good at math, both at 100%. And so what you will see, this is sometimes classic, when you start thinking about what makes you a great founder. Is you navigate what is apparent paradoxes. Another one I frequently talk about it, you got to be both flexible and persistent. And the reason for this is, entrepreneurs are frequently given the advise to have a vision, stay firm against your adversity. Realize that you have this vision that is contrarian to what people think and just stay on track. Get through the difficult times and get there. The other piece of advice given with each equal vigor is listen to data, listen to customers. Pivot, be flexible. Part of the thing this comes out to be in terms of being a great founder is to say well, when should I be persistent and when should I be flexible. And the vehicle I
most often use for this is you should have on a project you are doing, like a company, an investment thesis that essentially says why you think, possibly contrarian, why you think it is potentially a good idea. It should include what you know you think other people don't know.

And then as you are going into the battlefield, you go am I in fact increasing confidence in my investment thesis? Or decreasing value in my investment thesis? Because if I am increasing confidence then its hope to stay on track. Be persistent and by the way sometimes even with diversity you confidence can increase. If its decreasing that doesn't mean jump out. Paypal, LinkedIn, Airbnb, a whole bunch of startups I've been a part of have had months where you are like, Oh my god why did we ever think this was a good idea? Its kind of a valley in the shadows moment. For example, in PayPal it was August 2000 we were bring in twelve million in the expense curve was exponentiation, we had no revenue, decrease in confidence. However we say, what do we do in order to fix that and that gives you your immediate action plan.

Another one is, should I have belief or should you have fear? Should you essentially go, well i have this vision of the way the world should be an I should ignore everything else and I should just go with that. Well again, part of that being a great founder is, is being both able to hold the belief to think about where it is you want to be doing and want to be going. But also be smart enough that you are essentially listening to criticism, negative feedback, competitive entries. Where you are going, okay is this changing my investment thesis? Is this changing what I am planning on doing? It doesn't mean you lose confidence, you have the confidence but you also essentially have the parel. Again in this kind of thing is how do you put these two things together. Should I focus internally? Should I build a product, ignore the world, ignore competitors? Or should I focus externally, should I be recruiting? Should I be meeting people? Should I be gathering network intelligence? Again the answer is both. And the reason why I'm focusing on these type of habits, its both rather than either or. Is part of it makes a great founder is the ability to be flexible across these lines. To sometimes be 90% on way, sometimes be 80% the other way. Be executing the judgement on what does the current problem look like?

How is it that when I am trying to solve this that I should say this is what we should be doing and how should I be dividing the work? Part of when you think about these things is you say, this is another one that is classic, is people say well I am completely motivated by data its what customers say to user groups. I have a lot of entrepreneurial mythological, other kinds of things to talk about is gathering data, be guided on the data. Well actually in fact, data only exists in the framework of a vision that you are building, a hypothesis of where you are moving to. And the data can even be negative and you can think, well actually in fact this negative data means that I need to change or alter the way that I'm thinking about something. But i actually keep on a specific vision about what I'm doing. And by the way, sometimes when you have the specific vision you don't necessarily actually ever end up at that big vision that you were thinking about.

So for example, you know at Paypal we distributed these t shirts that said the new global world currency. Well actually in fact, I know Peter has been here, one of the jokes I told Peter was actually we do have this new world currency what we are trading is in dollars you may have heard of it, it existed for a while. Where essentially a mass merchant for that-- now of course this is message is what's happening with Bitcoin though thats a whole nother topic there. However, the key thing is is that vision saying we're creating a universal network that allows anyone to pay anyone to be a merchant to bring the electronics into the speed of commerce at any business that is being transacted. That vision kept a true north, but first we say well first we are going to have a banking model, then we think we are going to have a debt model, we are actually going to have a mass merchant model. How does that actually play out? So you are always combining the vision and the data, and dates within the framework of the vision. And sometimes the course of what you learn changes your vision.
Now this is actually one of the ones that I we save this special picture for one of the ones that I actually think is quite key.

Normally entrepreneurs founders, are thought about as having the risk takers. Where everyone else cowers in fear at this notion of risk, they boldly go out. Now thats true, you have to be a risk taker, you have to be thinking about how do I make a really coherent risk because in fact the only really big opportunities, the only contrarian opportunities smart people disagree with you on happen to be on that has more risks associated with them. On the other hand, part of the skill set, that when you are beginning to apply how you think about risks as an entrepreneur is how do I take intelligent risks? How do I take a focused risk, but if I'm right about that one thing then a bunch of other things break my way. And once I start doing that I try to figure out how to make my own shot possibility as high as possible? How do I minimize other risks? How do I essentially take this risk in an intelligent way that doesn't just go, oh yeah risk to the wind who cares but lets go. So this kind of combines that, this image is the best of the images that we found, is kind of the sense of that. Now back to what I was saying in terms of having an investment thesis.

Part of having a thesis is you chart it out its a list of bullets. For example with LinkedIn, everyone was going to be benefited by public professional network, everyone will realize including companies that its better to have it play out this way. The initial setup adoption will come from essentially people who visualize the world, play with it, and eventually the mass market will come on as they begin having network that is already having a network with value proposition to them. Thats what an investment thesis can look like and then you have economics, initially recruiting, and broadening those things.

You have that investment thesis, and you say is my investment thesis increasing or decreasing confidence? Do I think that the data that I get from the market ,when I talk to smart people, how does that change my confidence in it? This is how you minimize risks. For example, very early days in Paypal, part of what happened was they said they were going to cash and mobile phones with cash on Palm Pilots because its really easy. We actually realized the cash from Palm Pilots wouldn't work even before we launched the product. Basically what happened I went in and said to Max and Peter, I said here's our challenge--this group probably doesn't remember what Palm Pilots where, they were early PDA's. And so we lived in what was Palm Pilot central and the whole use case was splitting the dinner tab and everyone at table would have a Palm Pilots budget tab. Zero to one in every single restaurant. So you could, even by just thinking through the direction you are on you are going to hit a mine field and you need to pivot. And thats when Max Legend came up with the idea saying action packed sent by email. We can have email payments as the backbone of this and we were like yeah thats a good idea. Of course that what the whole thing pivoted into. And that is part of thinking though minimizing the risks as you are executing.

Heres another one thats kind of classic which is, should I have this long term vision or should I be solving a local near term problem? Again the answer is both these paradoxes. And the question is, you should jump between them. You should always have a long term vision in mind because if you actually completely lose your directions eventually you will find yourself somewhere in field thats not a good path out of. But if you are not focused on solving the problem thats immediately in front of you you're hosed. So part of the question about how to put these things together is you say, okay short term-what's the thing I need to be doing today? Have I made progress today? have I made progress this week? But is it largely on path? So I will give you an example of how this plays out in terms of financing or in terms of strategy. People frequently think product strategy is fundamental to how startups- I have a product idea, thats a thing, I'm a founder. Actually the next level down on strategy is usually product distribution and whether its consumer internet or enterprise, or anything because actually in fact no matter how good your product is if it doesn't get the customers you're hosed.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Lecture 13: The Information Age to the Networked Age: Are You Network Literate?

http://reidhoffman.org/information-age-networked-age-network-literate/

Reid Hoffman

It's said that when architects walk through an office, they see ceiling ornamentation, light sources, building acoustics. When psychologists walk through an office, they see unresolved father issues and avoidant personality disorders. When I walk through an office, I see networks. I know that makes me sound like the kid from *The Sixth Sense*. But I don't see dead people. I see networks.

When you truly see networks, it changes the way you plan and strategize. You move differently.

Take job hunting. The Networked Age has radically changed this activity, and yet when you ask people how they look for a job, a surprising number continue to say they "search the job listings." That's the Information Age approach! In the Networked Age, you should look for people with connections to companies you're interested in, trace the best path from those connections to people who can share useful intelligence, and then ask for introductions to those people.

Or consider investing. In my work at Greylock Partners, I don't form an investment theory and then go search for a startup that fits this theory. Nor do I purchase ad space in the Yellow Pages and hope that talented entrepreneurs let their fingers do the walking until they find me. Again, those are Information Age approaches.

The Networked Age approach? I focus on being a great ally to my network, and developing strong relationships where the information flow is highly reciprocal. I put myself at as many key intersections in my networks as I can. As a result, my network inevitably ends up connecting me with great entrepreneurs and great investments.

A decade ago, John Battelle stressed the importance of "search literacy." What he meant was that people who were skilled at using Google to find information had an edge over those who had yet to acquire this aptitude. In the Information Age, if you couldn't make sense of an increasingly information-rich world through effective search capabilities, you'd be culturally marginalized, just like a person who couldn't read street signs.

Now, those who can conceptualize and understand networks – both online and off – have an edge in today's fast-paced and hyper-competitive landscape, where the speed with which we can make informed decisions is critical. To wit, the subtitle of my forthcoming book is "Managing Talent in the Networked Age" — I think the networked age changes everything.

I like to use the word "literacy" in this context because it suggests a fundamental skill, a capability you must possess in order to effectively navigate the world. An illiterate person, a person who can't read street signs or complete job applications, has limited opportunities compared to others who possess these skills. A literate person moves freely and capably through the world.

So how do you know when you're network-literate? I think in terms of three levels that signify ascending competency:

**Apprentice:** Using network technology
**Journeyman:** Establishing a network identity
**Master:** Utilizing network intelligence
**Apprentice:** Using network technology

At this most basic level of network literacy, you're part of some networks. You have a Facebook profile, a LinkedIn profile, etc. You're using these networks to keep in touch with people you know, and on occasion, you may even use them to facilitate new connections.

While you may not be completely fluent yet, you understand that Facebook is more than just a place to announce what you had for lunch – it's a place to strengthen personal relationships. Similarly, you know that LinkedIn is more than just a repository for your digital resume. You use phrases and keywords with deliberate intention, to maximize your discoverability by the kinds of people you want to be found by.

In the case of my own LinkedIn profile, for example, my headline isn't "Executive Chairman of LinkedIn." It's "Entrepreneur. Product Strategist. Investor." That's because my LinkedIn profile is targeted primarily to entrepreneurs who might want financing from me.

(You'd be surprised at how many people simply use their current job title as the headline of their LinkedIn profile. This isn't wrong per se. But ultimately the headline on your LinkedIn headline is the first thing many people will see about you in a professional context – so it's an excellent place to choicefully craft your network identity. And your network identity is larger – or at least it should be larger — than your current position and company affiliation.)

Another way to make yourself more findable by the kinds of people you want to be found by are to join the same LinkedIn groups that they're participating in, or to follow relevant companies and individuals within the domain of your industry. Once you start thinking in terms of how the people you want to be found by might in fact find you – and tailoring your profile to maximize such potential discoveries – you have begun to think in a network-literate way.

**Journeyman:** Establishing a network identity

Once upon a time, we exercised unchecked authority over our identities, verbally air-brushing our resumes into highly
idealized portraits of ourselves, carefully vetting the references we chose to vouch for us. In the Networked Age, however, we’re all visibly and enduringly enmeshed in networks – even the so-called “self-made man” is a highly annotated specimen, with readily apparent links to the colleagues, mentors, institutions, and other entities that have helped shape the contours of his identity.

Indeed, we’re all the sum of an ongoing conversation that we initiate and propel, but which colleagues, customers, and even competitors contribute to as well. And while we once relied upon the broad strokes of resumes to define us, now we’re often judged by far more granular, network-derived metrics of influence and authority: Who retweets our tweets? Who comments on our Medium posts? Who shows up on LinkedIn as a 1 degree connection?

In the Networked Age, your professional identity expands well beyond your job title and the company you work for.

You’re not just “you” anymore. You’re also who you know, how they know you, what they know about you, who they know, and so on. At the Journeyman level, this way of thinking is becoming second nature to you. You understand that your identity is multivariate, distributed, and partially out of your control – your network helps shape your identity too.

Increasing your network literacy also means understanding other people’s network identities. Tell me the name of a person, and I’ll think of the network around them. I always see a person as part of a larger web of relationships. When I met Jeff Weiner, LinkedIn’s current CEO, I’d already had conversations with many of my own trusted colleagues about him. I had relationships with people that he had relationships with, and these strong points of network connectivity gave me a clear signal about Jeff and the kinds of people he trusted and valued most. I had a network portrait of him. And based on that portrait, I knew I wanted to build a strong relationship with him.

Master: Utilizing network intelligence

Spend five minutes watching your LinkedIn feed or Twitter timeline, and it’s clear that information proliferates even faster in the Networked Age than it did in the Information Age. Consequently, the ability to extract the right information at the right time is more crucial than ever. Search literacy is an important starting point, but in today’s high-velocity world, network literacy is increasingly crucial too.

In the Information Age, the New York Times, the Wall Street Journal, CNN, and eventually Google were typically people’s “first reads,” i.e., their default sources of new information and intelligence. Now, if you’re fully network-literate, your networks are your first reads – because you’ve consciously built up pipelines of people who reliably deliver information that is highly significant and relevant to you.

There is a whole “dark net” of critical-edge information that hasn’t made it into newspapers and blogs, information that exists only in people’s heads. In the past, such information was difficult to access for all but the best-connected and most persistent individuals. Now, it’s often just a few keystrokes away.

And if you’re fully network-literate, you’ve taken the time to understand the information flows within any given network. You know who the news breakers are. You know the thought leaders, the critics, and skeptics within a particular domain. You’re familiar with their preferred sources and biases.

While platforms like LinkedIn, Facebook, and Twitter certainly qualify as information Costcos, one-stop shopping for data en masse, the quality of your connections – and the strength of the relationships you have with them — generally matters more than the quantity. Ten extremely informed individuals who are happy to share what they know with you when you engage them can tell you a lot more than a thousand people you only know in the most superficial way.

But remember, using networks well is always a two-way street. People who exhibit the highest levels of network literacy know that the more relevant, high-quality information you share with others, the more such information you’re likely to receive. To be truly network literate is to always be thinking of how you can add value to the networks you’re a part of, and to make it a priority to turn connections into relationships, and relationships into alliances.

What Do You See When You Enter a Room?

These days, it’s not just Internet entrepreneurs who should see networks everywhere they look. When architects walk into a room, they should see networks. When psychologists walk into a room, they should see networks. In the Networked Age, we’re all like the little kid from The Sixth Sense. If you’re not seeing networks when you enter a room, you might want to check your pulse.

Learn how to support the development of network literacy inside your own company in my forthcoming book (with Ben Casnocha and Chris Yeh) titled The Alliance: Managing Talent in the Networked Age.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
At Greylock, my partners and I are driven by one guiding mission: always help entrepreneurs. It doesn’t matter whether an entrepreneur is in our portfolio, whether we’re considering an investment, or whether we’re casually meeting for the first time. Entrepreneurs often ask me for help with their financing decks. Because we value integrity and confidentiality at Greylock, we never share an entrepreneur’s pitch deck with others. What I’ve honorably been able to do, however, is share the deck I used to pitch LinkedIn to Greylock for a Series B investment back in 2004.

This past May was the 10th anniversary of LinkedIn, and while reflecting on my entrepreneurial journey, I realized that no one gets to see the presentation decks for successful companies. This gave me an idea: I could help many more entrepreneurs by making the deck available not just to the Greylock network of entrepreneurs, but to everyone. And so today I’ve published LinkedIn’s Series B deck on my personal website. There are three thematic emphases:

• how entrepreneurs should approach the pitching process
• the evolution of LinkedIn as a company
• the consumer internet landscape in 2004 vs. today

To help you figure out what aspects of the pitching process you’d like to understand better, I’ve summarized seven prevalent myths below, which I address more deeply in the full presentation.

1. **MYTH**: The startup financing process is about one thing — money.

   **TRUTH**: A successful financing process results in a partnership that delivers benefits beyond just money.

   A successful financing process obviously results in you raising capital for your company. But there are other critical outcomes you should shoot for as well. For example, great investors can significantly boost the strength of your network, which helps in recruiting employees and acquiring customers. Great investors can also be a source of network intelligence, so you can better prepare for likely challenges and opportunities ahead.

   Put another way, the ideal financing partner is a financing cofounder. This is why already-wealthy entrepreneurs raise money from experienced investors for their next startup: they know partnering with angels and venture capitalists is about more than just the money.

   Sadly, many investors actually add negative value, so an investor who adds no value (“dumb money”) but who doesn’t interfere with the operational process can sometimes be a
decent outcome. But ideally you find an investor who can proactively add value (“smart money”). How do you know if an investor will add value? Pay attention to whether they are being constructive during the pitch and financing process. Do they understand your market? Are their questions the same questions that keep you up at night? Are you learning from their feedback? Are they passionate about the problem you’re trying to solve?

2. **MYTH:** If your team is strong, show the team slide early in your pitch. **TRUTH:** Open your pitch with the investment thesis.

You have the most attention from investors in the first 60 seconds of your pitch, so how you begin is incredibly important. Most entrepreneurs start with a slide on the team. The team behind your idea is critical, but don’t open with that. Instead, open with what the investors have to believe in order to want to want to be shareholders in your company -- the investment thesis.

Your first slide should articulate the investment thesis in generally 3 to 8 bullet points. Then, spend the rest of the pitch backing up those claims and increasing investors’ confidence in your investment thesis -- which includes background on the team. Clearly articulate your investment thesis so investors can offer feedback that helps you refine it, eventually getting to a place where you both agree on it. This advice applies to seed funding rounds, too. Yes, seed investors understand that early stage companies have many unknowns and the idea will change a lot, so they look carefully at the people to see whether the team will be able to adapt. But even at this stage, lead with your overall investment thesis. Persuade investors your investment thesis is intriguing, then show who can make it happen.

3. **MYTH:** All investment pitches have the same structure. **TRUTH:** Decide whether your pitch is a data pitch or a concept pitch.

Your investment thesis is either concept-driven or data-driven. Which kind you are pitching?

In a data pitch, you lead with the data because you are emphasizing how good the data already is. Investors therefore evaluate your company based on the data. When LinkedIn went public, it was a data pitch to public market investors. We showed investors a multi-year track record of data.

If it’s a concept pitch, on the other hand, there may be data, but the data supports a yet undeveloped concept. A concept pitch shows your vision for how the future will be and how you will get to that future, so investors will want to buy a piece of it. Thus, concept pitches depend more on promised future data rather than present data.

4. **MYTH:** Avoid bringing up anything that might paint your business as risky and decrease investors’ confidence. **TRUTH:** Identify and steer into your risk factors.

Experienced investors know there are always risks. If they ask you about your risk factors and you can’t answer, you lose credibility because they assume you are either dishonest or dumb. Dishonest because if you’ve thought about the risk factors, but choose not to share them, you’re implying you’re not committed to a partnership. Dumb because you aren’t smart enough to understand that all projects have risk factors — including yours. Explicitly identify the one to three risks that could thwart your success and how you will mitigate them.

5. **MYTH:** Arguing that you have no prospective competitors is a strength. **TRUTH:** Acknowledge all types of competition and express your competitive advantage.

Entrepreneurs often say they have no competition, assuming that’s an impressive claim. But if you claim that you don’t have competition, you either believe the market is completely inefficient or no one else thinks your space is valuable. Both are folly.

The market is efficient, eventually -- if a valuable opportunity emerges, others will discover it. To build credibility with investors, you want to show that you understand the competitive risks and show why you’re going to win.

Express your competitive advantage this way: Why are you going to break out of the pack? What is your advantage? If you aren’t clear and decisive, investors won’t believe you have an edge that can lead to success.

6. **MYTH:** Don’t compare yourself to other companies because you think you’re unique. **TRUTH:** Pitch by analogy.

Every great consumer internet company grows up to be a unique organization. But in the early days, you want to use analogies to successful outcomes to describe what your company is and what its potential could be. Time is short -- it helps to refer to what those investors already understand. The best pitch I heard of was in Hollywood for a film called Man’s Best Friend. The pitch was “Jaws with Paws.” Investors were told that if the movie Jaws was a huge success, a similar plot but on land with a dog could also be a huge success. The movie turned out to be terrible, but the pitch was excellent.

To be sure, pitch by analogy but don’t necessarily reason by analogy. Reasoning by analogy, when you’re developing your business strategy, is dangerous. In startup land, you’re running across a minefield, so the details matter and you have to be careful with your analogies as you conceive strategy. But for high level pitches, analogies work great.

7. **MYTH:** Focus on today’s pitch. The future will take care
Every time you raise a round, you should be thinking about the subsequent round of financing. Assuming you successfully close the current round, how will you raise money later? Who will be the next investors you pitch? What will their concerns be? What will you need to solve next?

Expect that Series B investors will want to see some slides from your Series A deck. Series C investors will be similarly interested in your Series B deck. Etc. When I created our Series A deck, I presented a growth curve that would be good enough to get an investment, but I also had confidence that I could beat it. I wanted to be able to go into my Series B presentation and say, "Here’s what I said before, and here’s how I did." Because we beat our Series A expectations for network growth, investors could comfortably trust our promise to build revenue with our Series B financing.

Want to dive deeper and better understand how to pitch your startup? Read the full presentation at my personal website.

Last year, Ben Horowitz of Andreessen Horowitz articulated a well-thought-out philosophy on why he prefers to back Founder-CEOs and keep them in charge as the company grows. His essay, "Why We Prefer Founding CEOs" lays out three key ingredients that great founding CEOs tend to have, and that professional CEOs often lack:

- Comprehensive knowledge
- Moral Authority
- Total commitment to the long term

Ben’s point is that without these three key ingredients, a CEO won’t be able to maintain the rapid product innovation that is a prerequisite for success in today’s startup world. Ben cites Google and Cisco as rare exceptions where a professional CEO helped steer a company to market leadership and that the evidence is "one-sided and overwhelming" that you shouldn’t bring in a professional CEO. In other words, Ben asserts that bringing in a professional CEO should be a last resort for a founder.

And yet, many of the greatest success stories of the internet era involve founder/professional CEO partnerships. During the dot com era, Yahoo!’s Tim Koogle helped build Jerry Yang and Dave Filo’s startup into the world’s most valuable internet company. Meanwhile, Meg Whitman helped Pierre Omidyar’s eBay become the second most powerful ecommerce company in the world (trailing only Amazon). The Web 2.0 era provides successful examples like Joe Kennedy and Tim Westergren at Pandora, and the current social era provides even more, including Dick Costolo at Twitter and Tony Zingale and Dave Hersh at Jive.

Sure, there are plenty of cautionary tales about how VCs have ousted founders in favor of "professional CEOs" who run companies into the ground. But it’s hard to call the
Noam Wasserman of Harvard Business School has been studying what he calls “the founder’s dilemma” for nearly a decade. In his academic paper, "Rich versus King", he looked at 460 American startups from the 2000s. His statistical analysis showed that, paradoxically, founders maximized the value of their equity stakes by giving up the CEO position and board control: “The results show that, controlling for company size, age, and other factors, the more decision-making control kept (at both the CEO and board levels), the lower the value of the entrepreneur’s equity stake.”

In another study of 212 startups, Wasserman found that it was rare for Founder-CEOs to run their companies in the long term; less than half were still CEO after 3 years, and less than a quarter of the CEOs of the companies that reached an IPO were Founder-CEOs.

Given the evidence that bringing in an outside CEO can often pay off financially and, more importantly, in terms of overall scale and impact of the company, it's important to explore the possibility—even if it isn’t your first choice. I speak from personal experience, since I hired and partnered with a CEO, Jeff Weiner, very successfully at LinkedIn five years after co-founding the company.

Once you decide to evaluate the option of bringing in a professional CEO for your start-up, the real questions are, “Should I replace myself?” If the answer is “yes,” then “How do we make the transition?” And once you make the transition successfully, “How can I play a constructive long term co-founder role at the company?”

**How do I know when to replace myself?**

I love the early stages of building a company. The small team, building a brand-new product, out-innovating complacent incumbents...not only is the experience fresh and exciting, it also focuses on the things most founders love—especially technical ones: Solving interesting problems, developing new technologies, devising a unique strategy. But if you’re successful, the job of being CEO shifts dramatically over time. All of a sudden, you need to focus on a different set of challenges and concerns like establishing standard procedures and managing a large number of employees.

To remain successful, you have to be passionate about that kind of work as well. Ask yourself, “What am I focused on? What am I world-class at? What am I really committed to?” The answers will help you determine if you should bring in a CEO.

In my experience, CEOs need to derive satisfaction from the nuts and bolts of building a company, not just building product and articulating the vision. They need to be passionate about leadership, management, and organizational processes as the company scales.

To be a successful growth-stage CEO, you need to be ready to manage a 1,000 person organization and devote substantial time to time consuming things like running meetings and other business process. You can’t just do the exciting stuff like making the final call on product and speaking at conferences, while shuffling off everything else to the mythical COO who loves doing all the dirty work and doesn’t want any of the credit.

I went through this self-examination while I was still CEO of LinkedIn. I have always been passionate and committed, both to the company, and to its mission. I want to enable individual professionals to have more successful careers and to increase the productivity of mankind across entire industries and countries. I want a company that lives up to the original standard that Hewlett-Packard set for Silicon Valley—a great place for high-quality people to work that provides an experience that would continue to benefit them even after they move on to other things.

But as we scaled from a handful of people in my living room to dozens of people at an office, I saw the job of the CEO shifting. At 50 people and beyond, a CEO increasingly has to focus on process and organization, and that wasn’t what I was passionate about. For example, I didn’t like running a weekly staff meeting. I could do it, but I did so reluctantly, not enthusiastically. I’d rather be solving intellectual challenges and figuring out key strategies, not debating which employees should get a promotion, or configuring project timelines.

I co-founded LinkedIn in 2003, and by 2005, after asking myself the key questions about my passion, focus, and commitment, I knew I wanted to bring in a CEO. When I brought this up with my main VC, David Sze at Greylock, he had the same reaction I suspect I would have had: “Are you sure? Couldn’t we just hire a COO and have you stay as CEO?”

I had thought about the COO option, but I knew that the company needed someone who felt like they “owned the ball.” And I was confident I could partner well with a CEO, given my experience partnering with the various CEOs of companies being on the board. What’s more, the kind of person who has the capacity to be a great leader usually wants to be CEO, not COO. Sheryl Sandberg is one of the few great leaders who has been willing to be COO, and even then, she’s a unique COO.

At the time we began the search, I wouldn’t have believed how long the process would take.

After more than a year of searching, we brought Dan Nye in as LinkedIn’s new CEO at the beginning of 2007. Dan Nye came to us as CEO with a strong organizational background, having been a general manager at both Intuit and a different set of challenges and concerns like establishing standard procedures and managing a large number of employees. Given the evidence that bringing in an outside CEO can often pay off financially and, more importantly, in terms of overall scale and impact of the company, it's important to explore the possibility—even if it isn’t your first choice. I speak from personal experience, since I hired and partnered with a CEO, Jeff Weiner, very successfully at LinkedIn five years after co-founding the company.

Once you decide to evaluate the option of bringing in a professional CEO for your start-up, the real questions are, “Should I replace myself?” If the answer is “yes,” then “How do we make the transition?” And once you make the transition successfully, “How can I play a constructive long term co-founder role at the company?”

**How do I know when to replace myself?**

I love the early stages of building a company. The small team, building a brand-new product, out-innovating complacent incumbents...not only is the experience fresh and exciting, it also focuses on the things most founders love—especially technical ones: Solving interesting problems, developing new technologies, devising a unique strategy. But if you’re successful, the job of being CEO shifts dramatically over time. All of a sudden, you need to focus on a different set of challenges and concerns like establishing standard procedures and managing a large number of employees.

To remain successful, you have to be passionate about that kind of work as well. Ask yourself, “What am I focused on? What am I world-class at? What am I really committed to?” The answers will help you determine if you should bring in a CEO.

In my experience, CEOs need to derive satisfaction from the nuts and bolts of building a company, not just building product and articulating the vision. They need to be passionate about leadership, management, and organizational processes as the company scales.

To be a successful growth-stage CEO, you need to be ready to manage a 1,000 person organization and devote substantial time to time consuming things like running meetings and other business process. You can’t just do the exciting stuff like making the final call on product and speaking at conferences, while shuffling off everything else to the mythical COO who loves doing all the dirty work and doesn’t want any of the credit.

I went through this self-examination while I was still CEO of LinkedIn. I have always been passionate and committed, both to the company, and to its mission. I want to enable individual professionals to have more successful careers and to increase the productivity of mankind across entire industries and countries. I want a company that lives up to the original standard that Hewlett-Packard set for Silicon Valley—a great place for high-quality people to work that provides an experience that would continue to benefit them even after they move on to other things.

But as we scaled from a handful of people in my living room to dozens of people at an office, I saw the job of the CEO shifting. At 50 people and beyond, a CEO increasingly has to focus on process and organization, and that wasn’t what I was passionate about. For example, I didn’t like running a weekly staff meeting. I could do it, but I did so reluctantly, not enthusiastically. I’d rather be solving intellectual challenges and figuring out key strategies, not debating which employees should get a promotion, or configuring project timelines.

I co-founded LinkedIn in 2003, and by 2005, after asking myself the key questions about my passion, focus, and commitment, I knew I wanted to bring in a CEO. When I brought this up with my main VC, David Sze at Greylock, he had the same reaction I suspect I would have had: “Are you sure? Couldn’t we just hire a COO and have you stay as CEO?”

I had thought about the COO option, but I knew that the company needed someone who felt like they “owned the ball.” And I was confident I could partner well with a CEO, given my experience partnering with the various CEOs of companies being on the board. What’s more, the kind of person who has the capacity to be a great leader usually wants to be CEO, not COO. Sheryl Sandberg is one of the few great leaders who has been willing to be COO, and even then, she’s a unique COO.

At the time we began the search, I wouldn’t have believed how long the process would take.

After more than a year of searching, we brought Dan Nye in as LinkedIn’s new CEO at the beginning of 2007. Dan Nye came to us as CEO with a strong organizational background, having been a general manager at both Intuit and a...
and Advent, where he had responsibility for organizations many times the size of the early LinkedIn. In addition to strong capabilities, Dan is also perfect on integrity and culture. My idea when hiring Dan was that I could handle the product, and he could handle everything else. Dan helped us evolve LinkedIn from a product-focused startup into a complete company. During his time as CEO, he built a real sales department, rebuilt the executive team, and doubled the size of the company. But after a couple of years, I realized that as we continued to make major changes to the product, we needed a CEO who would “own” product as well.

My mistake was thinking you could divorce product strategy from the CEO role. It was a mistake related to the broken “grey-haired supervision” approach to professional CEOs. 20 years ago, you could count on product cycles lasting years, which meant that constantly developing new products and refining the vision was relatively less important than aggressive execution. The “professional” CEO back then just had to be a superb executor for the founder’s vision. The rise of internet time has reduced product cycles to months and weeks. As such, a CEO can’t focus solely on scaling concerns—today, the CEO has to be involved in the product.

So I decided to step back in temporarily as CEO and tried to find a new CEO with consumer internet product experience and organizational experience at scale. (Dan went on to continue being a very successful company builder, becoming the CEO of Rocket Lawyer and, among other successes, quadrupling its revenues.)

James Slavet, a partner at Greylock, introduced me to Jeff Weiner. James and Jeff had worked together at Yahoo!, and Jeff at the time was also an executive-in-residence at Greylock. As I got to know Jeff, I became convinced that he was the right choice to LinkedIn. I believed that Jeff’s experience at Yahoo! could help us by spreading a deep focus on consumer product insight and strategy throughout the entire company.

We named Jeff CEO of LinkedIn in the middle of 2009, about four years after I first approached David Sze about replacing myself.

How do we make the transition?

If it’s ideal for a CEO to have the knowledge, moral authority, and commitment of a founder, the answer is simple: Your transition process should bring the new CEO in as a co-founder of the company, not as an “adult supervisor.” Jack Dorsey has said as much: “Companies have multiple founding moments. I consider Dick Costolo to be a founding member of Twitter.” Dick wasn’t named CEO until 2010, four years after the company’s official start.

Think back to the list of all the successful Founder/CEO pairings. With almost no exceptions, four things were true:

1. The decision to step back from being CEO is a function of self-realization. It doesn’t work if the plan is being externally imposed by investors, for all the reasons Ben H. outlined. If you’re passionate about the nuts and bolts of building your company, and your VC simply thinks they know better, it’s probably a mistake to acquiesce.

2. The outside CEO was brought in early, so that he or she could play a real role in shaping the product, business, organization, and culture of the company. There is one exception to this pattern, which is when a company has lost its way, and the new CEO is essentially re-launching the company. The classic example here is Lou Gerstner, who transformed IBM from a failing hardware company to a services powerhouse.

3. The original team of founders was a small group of two to three people, making it easier to form strong co-founder bonds. Being able to build a trusting relationship is critical. In the cases of both Yahoo! and Google, Jerry/Dave/Tim and Larry/Sergey/Eric built warm and trusting relationships that lasted for years.

4. The new CEO had prior experience running a large organization. The whole point of hiring someone from the outside is to bring in skills and experience you don’t have, which will help scale the company. Here, Sheryl Sandberg represents the classic example—her management and people skills were brought in to complement Mark Zuckerberg’s great product vision and strategy. Notice how even as COO rather than CEO, Sheryl is accorded co-founder status, and is the lynchpin of the company’s management team and strategic decision-making.

Yahoo! presents a particularly fascinating example of these principles at work. When Tim Koogle came in as CEO in 1995, all four principles applied: The founders knew they wanted a CEO, Tim came in early, he formed a tight management triumvirate with the founders, and he had experience running large organizations from his time at Motorola. Later Yahoo CEOs failed to follow these principles, largely sidelining the founders until Jerry Yang’s return.

More recently, Marissa Mayer joined Yahoo at the Lou Gerstner phase—everyone acknowledges that Yahoo! needs to be remade. While her success is far from certain, if she does succeed, she will be viewed as a re-founder, not just a management caretaker.

My own experience bringing Jeff Weiner into LinkedIn stuck pretty close to these four principles.

1. As I’ve already detailed, the decision to bring in a professional CEO was one I initiated back in 2007, after a lot of self-examination.
Lecture 13: How To Be A Great Founder

2. While LinkedIn was already six years old when we brought Jeff in, it was still a relatively small company. Jeff was our 338th employee, and helped us launch our Talent Solutions business, which is now a key revenue driver for us.

3. Thanks to our strong mutual connections via Greylock, Jeff and I were able to bond and build a relationship both before and during the process of bringing him on to the team.

4. When Jeff was an EVP at Yahoo! he ran a 3,000-employee division. Not only was that far larger than LinkedIn at the time, it’s about as many employees as we have today!

What long term role can I play in our company?

I wanted to make sure that people made the shift from looking to me for answers to taking their cues from Jeff. Bringing in a CEO is like performing a brain transplant—you need to wire in a whole new set of connections. If the founder is still in the building, it’s all too easy for people to keep checking with him on every decision, rather than with the CEO.

When Jeff came in as CEO, I booked a hefty amount of travel for his first 6 months. When employees tried calling me to double-check a decision, I replied, “Sorry, I’m in Rome, talk to Jeff.” Jeff needed build up his own connectivity within the organization. By the time I returned from Europe, those connections were hard-wired.

For Jeff’s part, he went above and beyond to immerse himself in the company. For example, just a few months after Jeff joined LinkedIn, several engineers were sitting around at midnight in between bug fixes for what ultimately turned into a very late night product launch that extended into the wee early morning. One of the engineers decided to pull a graph of their new CEO’s login activity on LinkedIn.com. People were shocked: the only time period during the launch when Jeff was not consistently logged into the site was between 3:30 – 4:00 AM. It turned out he was obsessed with the product quality — just like a true founder.

To this day, Jeff is renowned for being one of LinkedIn’s most active users and is known for his ability to catch bugs before our developers.

Today, as Executive Chairman, my office at LinkedIn is next to Jeff’s, and when I’m not on the road or at Greylock, I’m on the LinkedIn campus most of the week. I enjoy helping the team on strategy, corporate and business development, and product vision. The most important thing I do, though, is sync up with Jeff every week about what’s on his mind. In our catch-up meetings, I’m able and willing to challenge his ideas. It’s hard for a CEO to get honest feedback and candid advice; so, provided you do not undercut him or her in the organization, as a founder you can play a uniquely helpful role in this respect. (And any CEO you hire should be eager to accept your honest counsel.) When it works, it makes the loneliest place in an organization—the CEO’s corner office—somewhat less lonely and potentially a lot more effective.

Of course, it might be that you’d rather transition out of a regular role after hiring a CEO. This can work, too. After bringing in Meg Whitman to run eBay, Pierre founded the Omidyar Network to make philanthropic investments, and returned to his native Hawaii where he’s done wonderful work for the community where he grew up. He’s still actively involved in eBay as chairman of the board, but he’s not at the office every week.

Conclusion

20 years ago, venture capitalists were in a hurry to bring in professional CEOs. Today, many of the same VC firms are busy touting their support for long-term Founder-CEOs. Both approaches can work, which means that as an entrepreneur, you should focus less on what's fashionable, and more on what’s right for you. This is a highly personal decision, and the right answer depends on you and your team—including your co-founders and your VCs. You might be a Steve Jobs, or you might be a Pierre Omidyar. As an investor, I’m willing to back you, even if you’re not sure which one you are yet. In every investment we make, we hope that the Founder-CEO will be able to lead the company to success, but if not, and if you realize as I did that you want to bring in a professional CEO, we’ll work with you to find someone who is a true partner.

So as it turns out, Ben was right. You always do want a Founder-CEO. But that person doesn't always have to be the Founding CEO. Being there at the start isn’t the only path to being a founder. “Founder” is a state of mind, not a job description, and if done right, even CEOs who join after day 1 can become Founders.
Lecture 13: Want to Start a Startup?

http://paulgraham.com/startupmistakes.html

Paul Graham

In the Q & A period after a recent talk, someone asked what made startups fail. After standing there gaping for a few seconds I realized this was kind of a trick question. It's equivalent to asking how to make a startup succeed—if you avoid every cause of failure, you succeed—and that's too big a question to answer on the fly.

Afterwards I realized it could be helpful to look at the problem from this direction. If you have a list of all the things you shouldn't do, you can turn that into a recipe for succeeding just by negating. And this form of list may be more useful in practice. It's easier to catch yourself doing something you shouldn't than always to remember to do something you should.[1]

In a sense there's just one mistake that kills startups: not making something users want. If you make something users want, you'll probably be fine, whatever else you do or don't do. And if you don't make something users want, then you're dead, whatever else you do or don't do. So really this is a list of 18 things that cause startups not to make something users want. Nearly all failure funnels through that.

1. Single Founder

Have you ever noticed how few successful startups were founded by just one person? Even companies you think of as having one founder, like Oracle, usually turn out to have more. It seems unlikely this is a coincidence.

What's wrong with having one founder? To start with, it's a vote of no confidence. It probably means the founder couldn't talk any of his friends into starting the company with him. That's pretty alarming, because his friends are the ones who know him best.

But even if the founder's friends were all wrong and the company is a good bet, he's still at a disadvantage. Starting a startup is too hard for one person. Even if you could do all the work yourself, you need colleagues to brainstorm with, to talk you out of stupid decisions, and to cheer you up when things go wrong.

The last one might be the most important. The low points in a startup are so low that few could bear them alone. When you have multiple founders, esprit de corps binds them together in a way that seems to violate conservation laws. Each thinks "I can't let my friends down." This is one of the most powerful forces in human nature, and it's missing when there's just one founder.

2. Bad Location

Startups prosper in some places and not others. Silicon Valley dominates, then Boston, then Seattle, Austin, Denver, and New York. After that there's not much. Even in New York the number of startups per capita is probably a 20th of what it is in Silicon Valley. In towns like Houston and Chicago and Detroit it's too small to measure.

Why is the falloff so sharp? Probably for the same reason it is in other industries. What's the sixth largest fashion center in the US? The sixth largest center for oil, or finance, or publishing? Whatever they are they're probably so far from the top that it would be misleading even to call them centers.

It's an interesting question why cities become startup hubs, but the reason startups prosper in them is probably the same as it is for any industry; that's where the experts are. Standards are higher; people are more sympathetic to what you're doing; the kind of people you want to hire want to live there; supporting industries are there; the people you run into in chance meetings are in the same business. Who knows exactly how these factors combine to boost startups in Silicon Valley and squish them in Detroit, but it's clear they do from the number of startups per capita in each.

3. Marginal Niche

Most of the groups that apply to Y Combinator suffer from a common problem: choosing a small, obscure niche in the hope of avoiding competition.

If you watch little kids playing sports, you notice that below a certain age they're afraid of the ball. When the ball comes near them their instinct is to avoid it. I didn't make a lot of catches as an eight year old outfielder, because whenever a fly ball came my way, I used to close my eyes and hold my glove up more for protection than in the hope of catching it.

Choosing a marginal project is the startup equivalent of my eight year old strategy for dealing with fly balls. If you make anything good, you're going to have competitors, so you may as well face that. You can only avoid competition by avoiding good ideas.

I think this shrinking from big problems is mostly unconscious. It's not that people think of grand ideas but decide to pursue smaller ones because they seem safer. Your unconscious won't even let you think of grand ideas. So the solution may be to think about ideas without involving yourself. What would be a great idea for someone else to do as a startup?

4. Derivative Idea

Many of the applications we get are imitations of some existing company. That's one source of ideas, but not the best. If you look at the origins of successful startups, few were started in imitation of some other startup. Where did they get their ideas? Usually from some specific, unsolved problem the founders identified.
Our startup made software for making online stores. When we started it, there wasn't any; the few sites you could order from were hand-made at great expense by web consultants. We knew that if online shopping ever took off, these sites would have to be generated by software, so we wrote some. Pretty straightforward.

It seems like the best problems to solve are ones that affect you personally. Apple happened because Steve Wozniak wanted a computer, Google because Larry and Sergey couldn't find stuff online, Hotmail because Sabeer Bhatia and Jack Smith couldn't exchange email at work.

So instead of copying the Facebook, with some variation that the Facebook rightly ignored, look for ideas from the other direction. Instead of starting from companies and working back to the problems they solved, look for problems and imagine the company that might solve them. (2) What do people complain about? What do you wish there was?

5. Obstinacy

In some fields the way to succeed is to have a vision of what you want to achieve, and to hold true to it no matter what setbacks you encounter. Starting startups is not one of them. The stick-to-your-vision approach works for something like winning an Olympic gold medal, where the problem is well-defined. Startups are more like science, where you need to follow the trail wherever it leads.

So don't get too attached to your original plan, because it's probably wrong. Most successful startups end up doing something different than they originally intended—often so different that it doesn't even seem like the same company. You have to be prepared to see the better idea when it arrives. And the hardest part of that is often discarding your old idea.

But openness to new ideas has to be tuned just right. Switching to a new idea every week will be equally fatal. Is there some kind of external test you can use? One is to ask whether the ideas represent some kind of progressions. If in each new idea you're able to re-use most of what you built for the previous ones, then you're probably in a process that converges. Whereas if you keep restarting from scratch, that's a bad sign.

Fortunately there's someone you can ask for advice: your users. If you're thinking about turning in some new direction and your users seem excited about it, it's probably a good bet.

6. Hiring Bad Programmers

I forgot to include this in the early versions of the list, because nearly all the founders I know are programmers. This is not a serious problem for them. They might accidentally hire someone bad, but it's not going to kill the company. In a pinch they can do whatever's required themselves.

But when I think about what killed most of the startups in the e-commerce business back in the 90s, it was bad programmers. A lot of those companies were started by business guys who thought the way startups worked was that you had some clever idea and then hired programmers to implement it. That's actually much harder than it sounds—almost impossibly hard in fact—because business guys can't tell which are the good programmers. They don't even get a shot at the best ones, because no one really good wants a job implementing the vision of a business guy.

In practice what happens is that the business guys choose people they think are good programmers (it says here on his resume that he's a Microsoft Certified Developer) but who aren't. Then they're mystified to find that their startup lumbers along like a World War II bomber while their competitors scream past like jet fighters. This kind of startup is in the same position as a big company, but without the advantages.

So how do you pick good programmers if you're not a programmer? I don't think there's an answer. I was about to say you'd have to find a good programmer to help you hire people. But if you can't recognize good programmers, how would you even do that?

7. Choosing the Wrong Platform

A related problem (since it tends to be done by bad programmers) is choosing the wrong platform. For example, I think a lot of startups during the Bubble killed themselves by deciding to build server-based applications on Windows. Hotmail was still running on FreeBSD for years after Microsoft bought it, presumably because Windows couldn't handle the load. If Hotmail's founders had chosen to use Windows, they would have been swamped.

PayPal only just dodged this bullet. After they merged with X.com, the new CEO wanted to switch to Windows—even after PayPal co-founder Max Levchin showed that their software scaled only 1% as well on Windows as Unix. Fortunately for PayPal they switched CEOs instead.

Platform is a vague word. It could mean an operating system, or a programming language, or a "framework" built on top of a programming language. It implies something that both supports and limits, like the foundation of a house.

The scary thing about platforms is that there are always some that seem to outsiders to be fine, responsible choices and yet, like Windows in the 90s, will destroy you if you choose them. Java applets were probably the most spectacular example. This was supposed to be the new way of delivering applications. Presumably it killed just about 100% of the startups who believed that.
How do you pick the right platforms? The usual way is to hire good programmers and let them choose. But there is a trick you could use if you’re not a programmer: visit a top computer science department and see what they use in research projects.

8. Silliness in Launching

Companies of all sizes have a hard time getting software done. It’s intrinsic to the medium; software is always 85% done. It takes an effort of will to push through this and get something released to users. [3]

Startups make all kinds of excuses for delaying their launch. Most are equivalent to the ones people use for procrastinating in everyday life. There’s something that needs to happen first. Maybe. But if the software were 100% finished and ready to launch at the push of a button, would they still be waiting?

One reason to launch quickly is that it forces you to actually finish some quantum of work. Nothing is truly finished till it’s released; you can see that from the rush of work that’s always involved in releasing anything, no matter how finished you thought it was. The other reason you need to launch is that it’s only by bouncing your idea off users that you fully understand it.

Several distinct problems manifest themselves as delays in launching: working too slowly; not truly understanding the problem; fear of having to deal with users; fear of being judged; working on too many different things; excessive perfectionism. Fortunately you can combat all of them by the simple expedient of forcing yourself to launch something fairly quickly.

9. Launching Too Early

Launching too slowly has probably killed a hundred times more startups than launching too fast, but it is possible to launch too fast. The danger here is that you ruin your reputation. You launch something, the early adopters try it out, and if it’s no good they may never come back.

So what’s the minimum you need to launch? We suggest startups think about what they plan to do, identify a core that’s both (a) useful on its own and (b) something that can be incrementally expanded into the whole project, and then get that done as soon as possible.

This is the same approach I (and many other programmers) use for writing software. Think about the overall goal, then start by writing the smallest subset of it that does anything useful. If it’s a subset, you’ll have to write it anyway, so in the worst case you won’t be wasting your time. But more likely you’ll find that implementing a working subset is both good for morale and helps you see more clearly what the rest should do.

The early adopters you need to impress are fairly tolerant. They don’t expect a newly launched product to do everything; it just has to do something.

10. Having No Specific User in Mind

You can’t build things users like without understanding them. I mentioned earlier that the most successful startups seem to have begun by trying to solve a problem their founders had. Perhaps there’s a rule here: perhaps you create wealth in proportion to how well you understand the problem you’re solving, and the problems you understand best are your own.[4]

That’s just a theory. What’s not a theory is the converse: if you’re trying to solve problems you don’t understand, you’re hosed.

And yet a surprising number of founders seem willing to assume that someone, they’re not sure exactly who, will want what they’re building. Do the founders want it? No, they’re not the target market. Who is? Teenagers. People interested in local events (that one is a perennial tarpit). Or “business” users. What “business” users? Gas stations? Movie studios? Defense contractors?

You can of course build something for users other than yourself. We did. But you should realize you’re stepping into dangerous territory. You’re flying on instruments, in effect, so you should (a) consciously shift gears, instead of assuming you can rely on your intuitions as you ordinarily would, and (b) look at the instruments.

In this case the instruments are the users. When designing for other people you have to be empirical. You can no longer guess what will work; you have to find users and measure their responses. So if you’re going to make something for teenagers or “business” users or some other group that doesn’t include you, you have to be able to talk some specific ones into using what you’re making. If you can’t, you’re on the wrong track.

11. Raising Too Little Money

Most successful startups take funding at some point. Like having more than one founder, it seems a good bet statistically. How much should you take, though?

Startup funding is measured in time. Every startup that isn’t profitable (meaning nearly all of them, initially) has a certain amount of time left before the money runs out and they have to stop. This is sometimes referred to as runway, as in “How much runway do you have left?” It’s a good metaphor because it reminds you that when the money runs out you’re going to be airborne or dead.

Too little money means not enough to get airborne. What airborne means depends on the situation. Usually you have to advance to a visibly higher level: if all you have is an idea,
a working prototype; if you have a prototype, launching; if you're launched, significant growth. It depends on investors, because until you're profitable that's who you have to convince.

So if you take money from investors, you have to take enough to get to the next step, whatever that is. Fortunately you have some control over both how much you spend and what the next step is. We advise startups to set both low, initially: spend practically nothing, and make your initial goal simply to build a solid prototype. This gives you maximum flexibility.

12. Spending Too Much

It's hard to distinguish spending too much from raising too little. If you run out of money, you could say either was the cause. The only way to decide which to call it is by comparison with other startups. If you raised five million and ran out of money, you probably spent too much.

Burning through too much money is not as common as it used to be. Founders seem to have learned that lesson. Plus it keeps getting cheaper to start a startup. So as of this writing few startups spend too much. None of the ones we've funded have. (And not just because we make small investments; many have gone on to raise further rounds.)

The classic way to burn through cash is by hiring a lot of people. This bites you twice: in addition to increasing your costs, it slows you down—so money that's getting consumed faster has to last longer. Most hackers understand why that happens; Fred Brooks explained it in The Mythical Man-Month.

We have three general suggestions about hiring: (a) don't do it if you can avoid it, (b) pay people with equity rather than salary, not just to save money, but because you want the kind of people who are committed enough to prefer that, and (c) only hire people who are either going to write code or go out and get users, because those are the only things you need at first.

13. Raising Too Much Money

It's obvious how too little money could kill you, but is there such a thing as having too much?

Yes and no. The problem is not so much the money itself as what comes with it. As one VC who spoke at Y Combinator said, "Once you take several million dollars of my money, the clock is ticking." If VCs fund you, they're not going to let you just put the money in the bank and keep operating as two guys living on ramen. They want that money to go to work. At the very least you'll move into proper office space and hire more people. That will change the atmosphere, and not entirely for the better. Now most of your people will be employees rather than founders. They won't be as committed; they'll need to be told what to do; they'll start to engage in office politics.

When you raise a lot of money, your company moves to the suburbs and has kids.

Perhaps more dangerously, once you take a lot of money it gets harder to change direction. Suppose your initial plan was to sell something to companies. After taking VC money you hire a sales force to do that. What happens now if you realize you should be making this for consumers instead of businesses? That's a completely different kind of selling. What happens, in practice, is that you don't realize that. The more people you have, the more you stay pointed in the same direction.

Another drawback of large investments is the time they take. The time required to raise money grows with the amount. When the amount rises into the millions, investors get very cautious. VCs never quite say yes or no; they just engage you in an apparently endless conversation. Raising VC scale investments is thus a huge time sink—more work, probably, than the startup itself. And you don't want to be spending all your time talking to investors while your competitors are spending theirs building things.

We advise founders who go on to seek VC money to take the first reasonable deal they get. If you get an offer from a reputable firm at a reasonable valuation with no unusually onerous terms, just take it and get on with building the company. Who cares if you could get a 30% better deal elsewhere? Economically, startups are an all-or-nothing game. Bargain-hunting among investors is a waste of time.

14. Poor Investor Management

As a founder, you have to manage your investors. You shouldn't ignore them, because they may have useful insights. But neither should you let them run the company. That's supposed to be your job. If investors had sufficient vision to run the companies they fund, why didn't they start them?

Pissing off investors by ignoring them is probably less dangerous than caving in to them. In our startup, we erred on the ignoring side. A lot of our energy got drained away in disputes with investors instead of going into the product. But this was less costly than giving in, which would probably have destroyed the company. If the founders know what they're doing, it's better to have half their attention focused on the product than the full attention of investors who don't.

How hard you have to work on managing investors usually depends on how much money you've taken. When you raise VC-scale money, the investors get a great deal of control. If they have a board majority, they're literally your bosses. In the more common case, where founders and investors are equally represented and the deciding vote is cast by neutral
outside directors, all the investors have to do is convince the outside directors and they control the company.

If things go well, this shouldn't matter. So long as you seem to be advancing rapidly, most investors will leave you alone. But things don't always go smoothly in startups. Investors have made trouble even for the most successful companies. One of the most famous examples is Apple, whose board made a nearly fatal blunder in firing Steve Jobs. Apparently even Google got a lot of grief from their investors early on.

15. Sacrificing Users to (Supposed) Profit

When I said at the beginning that if you make something users want, you'll be fine, you may have noticed I didn't mention anything about having the right business model. That's not because making money is unimportant. I'm not suggesting that founders start companies with no chance of making money in the hope of unloading them before they tank. The reason we tell founders not to worry about the business model initially is that making something people want is so much harder.

I don't know why it's so hard to make something people want. It seems like it should be straightforward. But you can tell it must be hard by how few startups do it.

Because making something people want is so much harder than making money from it, you should leave business models for later, just as you'd leave some trivial but messy feature for version 2. In version 1, solve the core problem. And the core problem in a startup is how to create wealth (= how much people want something x the number who want it), not how to convert that wealth into money.

The companies that win are the ones that put users first. Google, for example. They made search work, then worried about how to make money from it. And yet some startup founders still think it's irresponsible not to focus on the business model from the beginning. They're often encouraged in this by investors whose experience comes from less malleable industries.

It is irresponsible not to think about business models. It's just ten times more irresponsible not to think about the product.

16. Not Wanting to Get Your Hands Dirty

Nearly all programmers would rather spend their time writing code and have someone else handle the messy business of extracting money from it. And not just the lazy ones. Larry and Sergey apparently felt this way too at first. After developing their new search algorithm, the first thing they tried was to get some other company to buy it.

Start a company? Yech. Most hackers would rather just have ideas. But as Larry and Sergey found, there's not much of a market for ideas. No one trusts an idea till you embody it in a product and use that to grow a user base. Then they'll pay big time.

Maybe this will change, but I doubt it will change much. There's nothing like users for convincing acquirers. It's not just that the risk is decreased. The acquirers are human, and they have a hard time paying a bunch of young guys millions of dollars just for being clever. When the idea is embodied in a company with a lot of users, they can tell themselves they're buying the users rather than the cleverness, and this is easier for them to swallow. [9]

If you're going to attract users, you'll probably have to get up from your computer and go find some. It's unpleasant work, but if you can make yourself do it you have a much greater chance of succeeding. In the first batch of startups we funded, in the summer of 2005, most of the founders spent all their time building their applications. But there was one who was away half the time talking to executives at cell phone companies, trying to arrange deals. Can you imagine anything more painful for a hacker? [10] But it paid off, because this startup seems the most successful of that group by an order of magnitude.

If you want to start a startup, you have to face the fact that you can't just hack. At least one hacker will have to spend some of the time doing business stuff.

17. Fights Between Founders

Fights between founders are surprisingly common. About 20% of the startups we've funded have had a founder leave. It happens so often that we've reversed our attitude to vesting. We still don't require it, but now we advise founders to vest so there will be an orderly way for people to quit.

A founder leaving doesn't necessarily kill a startup, though. Plenty of successful startups have had that happen. [11] Fortunately it's usually the least committed founder who leaves. If there are three founders and one who was lukewarm leaves, big deal. If you have two and one leaves, or a guy with critical technical skills leaves, that's more of a problem. But even that is survivable. Blogger got down to one person, and they bounced back.

Most of the disputes I've seen between founders could have been avoided if they'd been more careful about who they started a company with. Most disputes are not due to the situation but the people. Which means they're inevitable. And most founders who've been burned by such disputes probably had misgivings, which they suppressed, when they started the company. Don't suppress misgivings. It's much easier to fix problems before the company is started than after. So don't include your housemate in your startup because he'd feel left out otherwise. Don't start a company with someone you dislike because they have some skill you need and you worry you won't find anyone else. The people are the most important ingredient in a startup, so don't compromise there.
18. A Half-Hearted Effort

The failed startups you hear most about are the spectacular flameouts. Those are actually the elite of failures. The most common type is not the one that makes spectacular mistakes, but the one that doesn't do much of anything—the one we never even hear about, because it was some project a couple guys started on the side while working on their day jobs, but which never got anywhere and was gradually abandoned.

Statistically, if you want to avoid failure, it would seem like the most important thing is to quit your day job. Most founders of failed startups don't quit their day jobs, and most founders of successful ones do. If startup failure were a disease, the CDC would be issuing bulletins warning people to avoid day jobs.

Does that mean you should quit your day job? Not necessarily. I'm guessing here, but I'd guess that many of these would-be founders may not have the kind of determination it takes to start a company, and that in the back of their minds, they know it. The reason they don't invest more time in their startup is that they know it's a bad investment. [12]

I'd also guess there's some band of people who could have succeeded if they'd taken the leap and done it full-time, but didn't. I have no idea how wide this band is, but if the winner/borderline/hopeless progression has the sort of distribution you'd expect, the number of people who could have made it, if they'd quit their day job, is probably an order of magnitude larger than the number who do make it. [13]

If that's true, most startups that could succeed fail because the founders don't devote their whole efforts to them. That certainly accords with what I see out in the world. Most startups fail because they don't make something people want, and the reason most don't is that they don't try hard enough.

In other words, starting startups is just like everything else. The biggest mistake you can make is not to try hard enough. To the extent there's a secret to success, it's not to be in denial about that.

Notes

[1] This is not a complete list of the causes of failure, just those you can control. There are also several you can't, notably ineptitude and bad luck.

[2] Ironically, one variant of the Facebook that might work is a facebook exclusively for college students.

[3] Steve Jobs tried to motivate people by saying "Real artists ship." This is a fine sentence, but unfortunately not true. Many famous works of art are unfinished. It's true in fields that have hard deadlines, like architecture and filmmaking, but even there people tend to be tweaking stuff till it's yanked out of their hands.

[4] There's probably also a second factor: startup founders tend to be at the leading edge of technology, so problems they face are probably especially valuable.

[5] You should take more than you think you'll need, maybe 50% to 100% more, because software takes longer to write and deals longer to close than you expect.

[6] Since people sometimes call us VCs, I should add that we're not. VCs invest large amounts of other people's money. We invest small amounts of our own, like angel investors.

[7] Not linearly of course, or it would take forever to raise five million dollars. In practice it just feels like it takes forever.

Though if you include the cases where VCs don't invest, it would literally take forever in the median case. And maybe we should, because the danger of chasing large investments is not just that they take a long time. That's the best case. The real danger is that you'll expend a lot of time and get nothing.

[8] Some VCs will offer you an artificially low valuation to see if you have the balls to ask for more. It's lame that VCs play such games, but some do. If you're dealing with one of those you should push back on the valuation a bit.

[9] Suppose YouTube's founders had gone to Google in 2005 and told them "Google Video is badly designed. Give us $10 million and we'll tell you all the mistakes you made." They would have gotten the royal raspberry. Eighteen months later Google paid $1.6 billion for the same lesson, partly because they could then tell themselves that they were buying a phenomenon, or a community, or some vague thing like that.

I don't mean to be hard on Google. They did better than their competitors, who may have now missed the video boat entirely.

[10] Yes, actually: dealing with the government. But phone companies are up there.

[11] Many more than most people realize, because companies don't advertise this. Did you know Apple originally had three founders?

[12] I'm not disissing these people. I don't have the determination myself. I've twice come close to starting startups since Viaweb, and both times I bailed because I
realized that without the spur of poverty I just wasn’t willing to endure the stress of a startup.

[13] So how do you know whether you’re in the category of people who should quit their day job, or the presumably larger one who shouldn’t? I got to the point of saying that this was hard to judge for yourself and that you should seek outside advice, before realizing that that’s what we do. We think of ourselves as investors, but viewed from the other direction Y Combinator is a service for advising people whether or not to quit their day job. We could be mistaken, and no doubt often are, but we do at least bet money on our conclusions.

Thanks to Sam Altman, Jessica Livingston, Greg McAdoo, and Robert Morris for reading drafts of this.
14. How to operate
I'm going to talk about how to operate. So I have watched some of the prior classes and I am going to assume you have hired a bunch of relentlessly resourceful people, you have built a product that at least some people love, hopefully raised some capital, and now you are trying to build a company. So you have been forging a product and now you are forging a company. And I would actually argue, forging a company is a lot harder than forging a product. Basic reason is people are irrational. We all know this. Either your parents, your significant other, your brother or sister, your teacher, somebody in your life is irrational. Building a company is basically like taking all the irrational people you know putting them in one building, and then living with them twelve hours a day at least. Its very challenging. Now there are some techniques for coping with that some people get good at it, some people don't. But that's really what operating is all about.

So basically what you are doing when building a company is building an engine. At first you have a drawing on a white board and you are architecting it, and it looks especially clean, beautiful, and pretty. But when you actually start translating it to practice it actually starts looking more like this and you're holding it together with duct tape. It takes a lot of effort from people to hold it together, that's why people work 80-100 hours a week. It's that heroic effort to keep this thing together because you don't actually, yet, have polished metal in place. Eventually you want to construct a very high performance machine. A machine that almost nobody really has to worry about every hour, every minute. And as we use to joke about eBay, that if the Martians took over eBay it would take 6 months for the world to notice. That's eventually what you want to get to.

As Warren Buffett says, build a company that idiots could run because eventually they will. So this is what you want. Basically a performance machine that idiots can run. Now as a leader, what is your real job, what's your role? Strictly speaking there is only one book ever written that actually explains how to do this. It's rather old, written in 1982 by Andy Grove, it's quite famous, and successful. And his definition of what your job is, is to maximize the output of the organization. Your organization that you are responsible for, so the CEO’s everything. VP would be a part of the organization, and the organizations around you. So if you are a VPE, you are actually responsible for the performance of the product team and the marketing team because you have influence there. So this is how you measure people, and you want to focus on the output and not the input. The old adage about measuring not measuring motion and confusing progress. You are measuring only progress. And this is going to sound like a fancy and glamorous thing to do. Maybe people get excited about managing a whole large organization and being “responsible” for the output. But in practice, what you are going to hopefully learn today is that it's more about things like ordering smoothies, teaching your receptionist how to answer your phone properly, and serving as a $10 an hour TaskRabbit for your employees. So let's talk about that.

So at first, when you start a company everything is going to feel like a mess. And it really should. If you have too much process, too much predictability, you are probably not innovating fast enough and creating enough. So it should feel like everyday there is a new problem and what you are doing is fundamentally triaging. So some things will look like a problem, and they are actually colds, they are just going to go away. So somebody is annoyed about this or that, that is a cold, you shouldn't stress about it and you certainly should not allocate all your time to it. And some things are going to present themselves as colds, but just like in the emergency room if they are not diagnosed properly they can...
actually become fatal. What I am going to try to do is give you frameworks for thinking about when things are colds and when things are potentially fatal.

So one of the most important things I learned at Square is the concept of editing. And this is the best metaphor I have ever seen in 14 years of running stuff, of how to think about your job. It's a natural metaphor, so it's easy to take with you everyday and it's easy to transmit to each of your employees so they can figure out if they are editing or writing. It's a natural construct, you generally know when somebody asks you to do something, am I writing or am I editing? So an editor is the best metaphor for your job. And we are going to talk about the specific things you are doing in editing.

The first thing an editor does and you have all probably had this experience in school, is you submit a paper, to a TA, a draft to your friend, and the first thing that editor does is they take out a red pen, or nowadays you go online, and they start striking things. Basically eliminating things, the biggest task of an editor is to simplify, simplify, simplify and that usually means omitting things. So that's your job too, is to clarify and simplify for everybody on your team. The more you simplify, the better people will perform. People can not understand and keep track of a long complicated set of initiatives. So you have to distill it down to one, two, or three things and use a framework they can repeat, they can repeat without thinking about, they can repeat to their friends, they can repeat at night.

Don't accept the excuse of complexity. A lot of people will tell you, this is too challenging, this is too complicated, yeah well I know other people simplify but that's not for me, this is a complicated business. They're wrong. You can change the world in 140 characters. You can build the most important companies in history with a very simple to describe concept. You can market products in less than 50 characters. There is no reason why you can't build your company the same way. So force yourself to simply every initiative, every product, every marketing, everything you do. Basically take out that red and start eliminating stuff.

Second thing editors do, is they ask you clarifying questions. When you present a paper to someone, what do they usually do? They find some ambiguity somewhere and ask, do you really mean this? Did you really mean that? Give me an example of this? That's what your job is. So you are in a meeting, people are going to look to you. And the real thing you do, is you ask a lot of questions. And they can be simple basic questions like should we try this seven days a week? Or six days? They can be fundamental questions like, where's our competitive advantage here? We try to do this as investors too. Some investors will ask you a billion questions about a billion things and they will have you do diligence forever. We try to narrow down to, what are the one two three four things that matter most to this company? And only focus on those things. So it allows us to be more decisive and we can make decisions rapidly. It allows us not to distract you from your day job which is actually building a company. And yet still I think we get to the highest fidelity question because we don't have all these extra extraneous details and data. Now it's hard, it's something you have to practice. But when you get good at it, every step you eliminate, Andy Grove estimated you can improve performance by 30-50%.

Now the next thing you do is you allocate resources. So the editor construct, this is what editors do all the time. They take editors from the Mideast, covering the Mideast and they move them to Silicon Valley, because Silicon Valley is more interesting. Or they move them to the sports section because they want to compete on the basis of sports journals and other publications. So that can be top down, where I take a whole bunch of resources and people and say, we are now going over here. We are going to compete on this basis. Then next month, next quarter, next year well that Middle East coverage is getting boring, we don't want to do that anymore. Let's go chase after something else. Or it can be bottom up, just like journalists mostly come up with their own stories.
The people who work with you, generally, should be coming up with their own initiatives. So a reporter, generally, who covers Google will come up with the interesting stories that they are hearing in ether and propose one or two to their editor for approval. But it’s not the editor saying, go cover Google and this is the angle I want. Once in a while they do that, but its not the meat and potatoes of what a journalist does every day. Your goal over time is to use less red ink every day. So one way of measuring how well you are doing at communicating or talking to your colleagues about what’s important and what’s not, about why some things are important and why some things are not. It’s how much red ink you are pulling out in a day, it’s okay if you are having a bad day and the red ink is all over the place. But it’s not okay if the red ink next month is more than it was last month, and next quarter more than this, so measure yourself by how much red ink you’re creating.

The other thing that is very important that actually isn’t as intuitive to a lot of people, is the job of an editor is to ensure consistent voice. So if any of you read The Economist, many of you can tell that there is one consistent voice. You can pick up any article, any post in The Economist and it feels like it was written by the same person. Ideally, your company should feel, on your website, on PR releases, on your packaging if it’s a physical product, anywhere on your recruiting pages it should feel like it was written by one person. That’s extremely difficult to do. And at first you are going to be tempted to do that yourself, which is okay for a founder to do that him or herself initially. Over time you do not want to be doing all of the consistent voice editing by yourself. You want to train people so they can recognize differences in voice.

See this website page, it looks very different than this recruiting page. You start asking questions, why is that? Is the reporting messed up? Is one of the leaders over here not really understanding the voice of the company? You have to fix that over time, but you want to start with the objective that everything should feel the same. It’s quite difficult to practice, almost every company has one piece of the organization that isn’t on the same voice. At Apple, which is notorious, even under Steve’s regime, which is notorious for getting this right, if you asked someone who worked at Apple, asked them about the internal tools about recruiting, do they really feel like Apple products? All of them will tell you no. So you are never 100%. But you definitely want to get as close to that as you can.

Next complicated topic is delegating. So just like the other metaphor on editing is writers do most of the work in the world, editors are not writing most of the content in any publication. So that is true of your company, you shouldn’t be doing most of the work. And the way you get out of most of the work, is you delegate. Now the problem with delegating is that you are actually responsible for everything. The CEO, founder, there is no excuse. There is no, there is that department over there, this person over there screwed up. You are always responsible for every single thing, especially when things go wrong. So how do you both delegate but not abdicate? It’s a pretty tricky talent, both are sins. You over delegate and you abdicate, or you micromanage, those are both sins. So I’m going to give you a couple techniques for solving this.

First, and this actually came from High Output Management and Andy Grove, is called task relevant maturity. It’s a fancy phrase for, has this person ever done this before? It’s really simple, how mature is this person in doing something? And the more they have done the exact same task before, the more rope you are going to give them. And the more they are trying something new, the more you are going to instruct them and constantly monitor. This is a basic concept but it’s worth keeping in the back of your brain. The interesting implication, and this is pretty radical, is that any executive, any CEO, should not have one management style. Your management style should be dictated by your employee. So with one particular person, you may be very much a micromanager because they are quite low on this scale. And with another person,, you may be delegating a lot because they are quite mature on this scale.
So it's actually a good thing if you do reference checks on somebody and half the people you call say they are a micromanager and the other half say they actually give me a lot of responsibility. That's a feature not a bug. I didn't understand that at first at all. I used to be befuddled when people would do reference checks on me and come back with this complicated mosaic. Then I finally figured out that maybe I was doing my job correctly. So then I taught others that this is the way to do it.

A more nuanced answer that I came up with, is how to make decisions. Delegating vs doing it yourself. You don't want to do it yourself too often. So I basically borrowed from Peter, this is my first two by two matrix ever in my life, but he taught me something at least. You basically sort your own level of conviction about a decision on a grate, extremely high or extremely low. There's times when you know something is a mistake and there's times when you wouldn't really do it that way but you have no idea whether it's the right or wrong answer. And then there is a consequence dimension. There are things that if you make the wrong decision are very catastrophic to your company and you will fail. There are things that are pretty low impact. At the end of the day they aren't really going to make a big difference, at least initially.

So what I basically believe is where there is low consequence and you have very low confidence in your own opinion, you should absolutely delegate. And delegate completely, let people make mistakes and learn. On the other side, obviously where the consequences are dramatic and you have extremely high conviction that you are right, you actually can't let your junior colleague make a mistake. You're ultimately responsible for that mistake and it's really important. You just can't allow that to happen. Now the best way to do that is to actually explain your thinking why. It's easy to shortcut when you get busy explaining ways in the world but it's very important to try.

When I was at LinkedIn, I had a colleague that was quite, quite talented but occasionally would get annoyed if I did not agree with his opinion on something. So I would spend a lot of time trying to persuade him why I was making a decision a certain way. And his wild card, his card he would call out if I didn't quite persuade him was, okay you're the boss. And that to me was like I was burning a lot of social capital. Every time he said that I knew I was creating a really thin line and ultimately that was going to backfire if I did that too often. You want to track the times that you are doing that.

An example of this is at Square, one of my favorite people in the world and my second hire, first marketing hire, had this program he wanted to run called Inner Square which allowed Square merchants to give out, imagine a food truck outside put out ten Squares on the counter and people could just grab them. And Kyle had this great idea that this would be an awesome marketing program. Squares would spread Squares to other people and to some extent it was on brand. So it didn't have catastrophic consequences. Each of these ten Squares didn't cost that much money, so financially we could afford to do it. But at that time, my ten years of experience said it was not going to work on a meaningful enough scale for our metrics and I preferred not to do it. Kyle was so excited about this that I decided to just let him do it. He learned that when you measure this thing, it's not massive. It doesn't create massive value for the company. It did require a fair amount of operational complexity to ship all these Squares to people and figure out how to get them, etc, etc. But it allowed him to be excited about his job and to learn how to filter future ideas. So it was totally worth letting him make the "Mistake."

The next and most important thing you do is edit the team. So these are the people you work with. Nobody is going to have a perfect team and you certainly aren't going to start that way. So what I am going to try to do is maximize the probability of success in editing the team. So I like this idea of barrels and ammunition. Most companies, when they get into hiring mode, as Sam pointed out you should defer that a bit, but when you do just hire a lot of people, you expect that when you add more people your horsepower or your velocity of shipping things is going to increase. Turns out it doesn't work that way. When you hire more engineers you don't get that much more done. You actually
sometimes get less done. You hire more designers, you definitely don't get more done, you get less done in a day.

The reason why is because most great people actually are ammunition. But what you need in your company are barrels. And you can only shoot through the unique barrels that you have. That's how the velocity of your company improves is having barrels. Then you stock them with ammunition, then you can do a lot. You go from one barrel company, which is mostly how you start, to a two barrel company and you get twice as many things done in a day, per week, per quarter. If you go to three barrels, great. If you go to four barrels, awesome. Barrels are very difficult to find. But when you have them, give them lots of equity. Promote them, take them to dinner every week, because they are virtually irreplaceable because they are also very culturally specific. So a barrel at one company may not be a barrel at another company.

One of the ways, the definition of a barrel is, they can take an idea from conception and take it all the way to shipping and bring people with them. And that's a very cultural skill set. Two questions are probably occurring to you. How can you tell who is a barrel and who is not? One is you start with a very small set of responsibilities, it can be very trivial. It can be something like, I want to reward the engineers in my office at nine o'clock every night with a nice cold, fresh smoothie. This is actually a real example. I was frustrated, our engineers were working really hard, and maybe 20%, 30% would stay late in the evening and we had already served them dinner but I wanted to give them something cool to reward them. You can think about alcohol but that's a little complicated. So smoothies were probably a little bit better than pizza, which drains you of energy. But nobody could get smoothies to show up in my office at nine o'clock sharp, that were cold, that tasted good, and that were delivered in the right place that the engineers would find them.

You would think this is simple but in fact it took two months to get this done. So we had an intern start, and I think on his second day I was explaining this problem, and he said, well I will do it. And I was looking at him like there was no way. I have seen my office manager fail, my assistant fail, who were actually pretty good. This just isn't going to happen. And low and behold they show up. On time, cold, delivered at the right place, and my first instinct was great. Nothing about the smoothies, but now I can actually give him something more important that is more complicated to do.

And that's actually what you want to do with every since employee, every single day, is expand the scope of responsibilities until it breaks. And it will break, everybody, I couldn't run the world, everybody has some level of complexity that they can handle. And what you want to do is keep expanding it until you see where it breaks and that's the role they should stay in. That level of sophistication. But some people will surprise you. There will be some people that you do not expect. With different backgrounds, without a lot of experience that can just handle enormously difficult tasks. So keep testing that and pushing the envelope. The other signal to look for is once you've hired someone, with an open office, just watch who goes up to other people's desks. Particularly people they don't report to. If someone keeps going to some individual employees desk and they don't report them, it's a sign that they believe that person can help them. So if you see that consistently, those are your barrels. Just promote them, give them as much opportunity as you can.

The other question everybody asks about people is when do you hire somebody above somebody. And when do you mentor somebody, and when do you replace somebody? And the way to think about this is that every company has their own growth rate, and every individual has their own growth rate. So some companies that are very successful, lets say LinkedIn. LinkedIn was always a very linear company, it never went like this. So for example I joined LinkedIn 18 months after we launched and we only had 1.5 million users. Which for a social product is a very small number. And when I joined I was the twenty-seventh employee, and when I left two and a half years later they only had fifty-seven employees. In
contrast, when I joined Square as the twentieth employee, two and a half years later we had two hundred fifty-three employees.

So each company has its own velocity on this curve. So if the company is going like this, you can only keep people on the roles if their own learning curve is going like this. On the other hand, if your learning curve is like this, anyone learning faster than that, you can give them the same roles as they do. So always track the individual slope of employee and the company growth rate. Now that you have your barrels figured out, and you can identify people who can take ideas that you have in the back of your head, scope it out, run with it, ship it, and it's perfect. Where do you aim these barrels?

So I am going to argue that you need to spend a lot of time focusing on people. This is something I learned from Peter Thiel actually. He used to insist at PayPal that every single person could only do exactly one thing. And we all rebelled, every single person in the company rebelled to this idea. Because it's so unnatural, it's so different than other companies where people wanted to do multiple things, especially as you get more senior, you definitely want to do more things and you feel insulted to be asked to do just one thing.

Peter would enforce this pretty strictly. He would say, I will not talk to you about anything else besides than this one thing I assigned you. I don't want to hear about how great you are doing over here, just shut up, and Peter would run away. And then focus until you conquer this one problem. And the insight behind this is that most people will solve problems that they understand how to solve. Roughly speaking, they will solve B+ problems instead of A+ problems. A+ problems are high impact problems for your company but they are difficult. You don't wake up in the morning with a solution, so you tend to procrastinate them. So imagine you wake up in the morning and create a list of things to do today, there's usually the A+ one on the top of the list, but you never get around to it. And so you solve the second and third. Then you have a company of over a hundred people so it cascades. You have a company that is always solving B+ things which does mean you grow, which does mean you add value, but you never really create that breakthrough idea. No one is spending 100% of their time banging their head against the wall every day until they solve it. So I highly recommend some version of that. You can be less stringent, you can give people three things to work on, but I would still track the concept of what would happen if you only gave everybody one thing to prioritize.

You don't want to be making all these decisions yourself. You have to create tools that enable people to make decisions at the same level you would make them yourself. So how do you create scale and leverage? The first thing I would recommend is to build a dashboard. This is an old Square dashboard, it looks pretty presentable even today. The construct of the dashboard should be drafted by the founder. You need to simplify the value proposition in the company's metrics for success on a whiteboard. You can have other people build the dashboard, I don't care about that. But you need to draw it out. Like what does business success look to us and key inputs to those and then have someone create something that is very intuitive for every single person in the company, including customer support to use. And then, the key metric of whether you succeed is what fraction of your employees use that dashboard every day? If it's actually useful, it should be close to 100%. It's not going to be probably 100% but you want to measure that. Just like you have quality scores for your other KPI's with users, your dashboard needs to be as intuitive as your product is for users.

Another concept is transparency. Transparency people talk a lot about, it's a goal everybody ascribes to but when push comes to shove, very few people actually adhere to it. So let me walk through a little bit of transparency and different stages of transparency. Metrics are the first step. So everyone in your company should have access to what's going on. Other things I like to do, is to take your board decks. As you get more formal, the board decks will get more complicated. And actually review every single slide with every single employee after the board meeting. You can strip out the compensation information if you really want to. But every other slide you should go through with every single
employee and explain it. If you can remember some of the feedback you got from your board that is really cool to pass on.

Another thing we did at Square is when we scaled, not everybody is going to get invited to every meeting, but they are going to want to go to every meeting. The way you scale that is you create notes for every meeting and you send them to the entire company. So we created notes at least for every single meeting that involved more than two people, somebody would write notes and send it to the entire company. So everyone felt that as the company added employees, they continued to track what was interesting, what was going on. And they never felt excluded, hopefully. Another thing is, even details around conference rooms. Every conference room at Square has glass walls. Because as soon as you have regular walls, people wonder what's going on. It's amazing, if they can see who exactly is in the meeting and who is meeting with who when, they don't start to worry nearly as much as about what's going on behind those closed doors.

Stripe, you may have seen a blog post about, I think Patrick wrote it, about email transparency, about actually allowing everyone to have access to email. That's pretty far out there but it has certain merits to it. I would call all the tactics you hear and read about as minimal viable transparency. I actually think you could push the envelope a lot more. Steve Jobs actually tried this at Next, he actually tried transparent compensation. I actually think that even though Next didn't do extremely well, the real reason wasn't around experimenting with compensation in transparency. There is a lot of merit to that. The critique of compensation transparency is, well we want people to be teammates and work together and collaborate. And if you look in the sports world where people are teammates and collaborate, all of their compensation is actually public. In fact, any one of us can look up one of their compensations in the sports world to get it exactly accurate. And somehow it seems to work. So I am not completely bought into that you need to keep compensation transparent.

And finally, metrics. So you want to measure things. You want to measure outputs, not inputs. And again, you should dictate this yourself. You should draft the dashboard yourself to tie this all together. One important concept is pairing indicators. Which is, if you measure one thing and only one thing, the company tends not to optimize to that. And often at the expense of something that is important. Cost is example of payments and financial services is risk. It's really easy to give the risk team the objective and say, we want to lower our fraud rate. It sounds great. Until they start treating every user in this audience as a suspect because they want to lower the fraud rate. So they require each of you to call them up on the phone and give them more supplemental information and fax in things. Then you have the lowest fraud rate in the world, you also have the lowest level of customer satisfaction score.

What you want to measure at the same rate as your fraud rate, is your false positive rate. That forces the team to actually innovate. Similarly, you can give recruiters metrics around hiring. And guess what? You will have a lot of people come in for interviews. But if you are not tracking the quality of interviewers, you may be very unhappy about the quality of people you are hiring and giving interviews to. So you always want to create the opposite and measure both. And the people responsible for that team need to be measured on both.

Finally around metrics. One insight I have had over my career is what you, you kind of want to look for the anomalies. You don't actually want to look for the expected behavior. So a famous example was at PayPal. None of the top ten markets that the company was planning on going after included eBay. One day, someone noticed that 54 of the sellers actually handwritten into their eBay listings, please pay me with Paypal and brought this to the attention of the executive team at the time. The first reaction from the executive team was, what the hell is going on? Let's get them out of the system, that is not the focus. Fortunately, David Sacks came back the next day and said, I think we found our market. Let's actually build tools for these power sellers instead of forcing them to write into their listing, pay me.
with PayPal. Why don't we just have an HTML button that they can just insert? And that actually worked. Then he thought, why should we have them insert it each time? Why don't we just automatically insert it for them? They can just insert it once, then every listing they have forever will have it automagically appear there. So that became the success for PayPal. Similarly I was at LinkedIn and I saw this stat that made no sense to me. The UI of the site was a little different then. 25% of all clicks, maybe 35% of all clicks from the homepage were people going to their own profile. And that didn't make any sense whatsoever. It was in the settings, you had to go to the margin and find a link. It was 25-35% of every click at scale, so this is just invalid stuff. And it made no sense whatsoever. I had never seen UI perform that way.
Lecture 14: Readings

- Bill Walsh, The Score Takes Care of Itself, pp. 2-31, 137-146, 202-203
- Andy Grove, High Output Management, Chapters 3 (optional), 4, 9, 11, 13, 14

https://archive.org/details/HighOutputManagement_201308
15. How to manage
Lecture 15: How To Manage

http://startupclass.samaltman.com/courses/lec15/

Ben Horowitz

When Sam originally sent an email for me to do this course, he said "Ben can you teach a fifty minute course on management?" And I immediately thought to myself, "Wow, I just wrote a three hundred page book on management. So that book was entirely too long."

I didn't actually have time to collapse the three hundred pages into fifty minutes. Like Mark Twain, I didn't have time to write a good short letter, so I'm going to write a long letter. But in this case, I am going to teach exactly one management concept.

This management concept is the thing that I see CEO's mess up more consistently then anything else and from when they're very, very early to when they're very, very big as a company. It's the easiest thing to say and the most difficult to master. The concept in musical form is from Sly and the Family Stone. "Sometimes I'm right and I can be wrong. My beliefs are in my song. No difference what group I'm in."

So that's the musical version of today's lesson. For those of you who are musical you can leave now.

When you're making a critical decision, you have to understand how it's going to be interpreted from all points of view. Not just your point of view and not just the person you're talking to but the people who aren't in the room, everybody else. In other words, you really have to be able when making critical decisions to see the decision through the eyes of the company as a whole, which means you have to add up every employee's view and then incorporate that into your own view. Otherwise your management decisions are going to have very weird side effects and potentially very dangerous consequences. It's a really hard thing to do because at the point when you are making a decision you're often under a great deal of pressure.

Let's get into the agenda. I am going to cover kind of four cases. First, I am going to cover demotions, which a very emotional thing. Then raises, which is also an emotional thing. Then we are going to evaluate one of Sam's blog posts, which is news to Sam. I figured I'd tease since he invited me to do a fifty minute management class after I wrote a three hundred page book.

Then I'm going to talk about history's greatest practitioner at this, I'm wearing a shirt with him on it, and how he used it to do something that nobody had ever done and has ever done since in human history. Basically complete mastery of the technique I am going to talk about.

So, first business example, you've got an executive, and do you demote or do you fire him? This comes from an actual conversation, an actual real life situation that I was working on with a CEO. The basic situation was this: he had a great executive he had hired. He was working harder than anybody else in the company and was doing everything he was supposed to. Everybody liked him because he worked so hard and was a generally a smart person, but he was in over his head knowledge-wise. He did not have the knowledge and the skills to do what the company needed him to do or really compete against the competition. So he couldn't actually keep him in the job, but he was a great guy. So the question is, should I fire this person or can I just move him into a lower role and bring in a person above him. That would be cool. Let's look at how you make that decision.
You are, in this case, the CEO. It's really hard if somebody comes to work every day at six AM, is working until ten PM, and is working harder than anybody in the company. It's really hard to just say, "Well sorry, nice effort but you don't get an A for an effort. You get an F because I fired you."

Nobody wants to have that conversation. A demotion is kind of neat because from a CEO's point of view, he can keep him in the company. He works so hard. He's a great example of somebody who gives great effort. He's got a lot of friends in the company, so from a cultural standpoint it's a win/win because he gets to stay. Then I can bring in somebody who can solve my problem but I don't have to create another problem.

If you think about it from the executive's perspective, it's like, "I don't want to be demoted but I really don't want to be fired because if I get fired, that's a way harder more complicated thing to explain to my next employer that I got demoted. Getting demoted is, well I didn't really get demoted. I got a new job, a smaller title."

The last thing is it enables, theoretically: the company values all of our employees. We brought you in. We made a commitment to you as an employee and it will enable you to keep growing with the company. This was the conversation I was having with the CEO and I said, "Well, wait a minute. Let me ask you this. What's the equity package that this executive has?" He goes, "Well, what do you mean?" I was like, "Well I would like to see direct level of compensation? Does he get a Vice President level of compensation? Does he have a point one percent? Does he have point four percent?"

That gave the CEO pause. He's like, "Well, he does have a point in a half." And I was like, "Ok. So you're an Engineer in your company. How do you feel about somebody who used to be the head of sales who brought in with a point and a half? What do your Engineers get? Do they get point one percent? Point two percent? What are they getting at this point? How are they going to feel about somebody who is NOT the Head of Sales with one point five percent of the company?" And he was like, "Uh oh." And I was like, "Yeah! Uh oh. Because how fair is that? Are you going to take the equity away? Are you up to do that? Are you up to go back and take back his compensation? How productive do you think he'll be if you take away his compensation? Secondly, will people give him the same respect now that you've demoted him? Because they knew him as this and now he's this. "I knew you when you were Head of Sales now you're the Regional Manager and you're telling me what the f*** to do? You're telling me I need to make that call? It seems to me you got demoted. Who are you to talk to me? I am the up and comer. I am going to be the next VP of Sales at the company."

All these things come into play. When you look at the end, you may think you are dealing with one person. You may think that this is a demotion or a firing of one person. What does it mean according to that one person? But what you are really doing is saying, what does it mean to fail on the job? Particularly the highest paid, the highest compensated job in the company from an equity standpoint. And then, what's required to maintain your equity? Is it good enough to put in an effort or do you have to get a result? In different situations at different levels, the answers will be different. If this had been a person who was not an Executive brought in from the outside, but someone you promoted past where they should have been and didn't ever get that equity, maybe you make a different decision but you have to understand what it's going to mean to everybody, not just the person you're talking to.

Example two: An excellent employee asks for a raise. A good employee, this isn't like the last employee. First thing you think is they're really good, they asked me for a raise, they didn't ask me for no reason. They asked me because they think they deserve it. I want to retain them. I want to be fair! They've done a great job. I know that if I give them this raise, it's going to be all love coming my way. If I give you a raise we're good. You got a raise! It's awesome.
Um. So like you know from your perspective you know what you want to do when somebody asks you for a raise. And then if you look at it - ok of for - what about from their perspective? How would they like take if you gave them the raise. You have to remember the Employee from - for them to get to point where they've ask you for a raise this is not something they just like woke up one morning and like I am going in and asking him for a reason. Like this is something where they've thought about it a lot. They've compared their other options. They may have an offer from another company. You know it's something their spouse probably has been talking to them about. And so it's a serious thing. Um, uh, so if you give it to them they're very likely to feel very good about it. Like they may be like paranoid but like " Why you giving me a raise?" But very unlikely. Much more they'll feel like...

(Plays "They Do the Schmoney Dance" video)

**Ben Horowitz:** Uh, for those who don't know - that's Bobby Schmurda and Rowdy Rebel doing the Schmoney Dance. Um, but that's like the reaction you'll get. So there is a lot of momentum to say, “Yes look. You know they've read Cheryl's book. They've learned in and I'm going to reward them.”

**Ben Horowitz:** For doing all that. Which is, you know and by the way that book is very good advice. So I'm not knocking Cheryl on that. I don't want you to be misinterpreting me. Um, however (Ben Horowitz chuckles) you know there was going to be a however. What about - you have to think about it through the point of view of the Employee who did not ask for the raise. So, the Employee who didn't ask for the raise they may be doing a better job than the Employee who did ask for the raise. And in their mind they are going, “Ok, so I didn't ask for a raise and I didn't get a raise. And they asked for a raise and they got a raise. And so what does that mean?” That means like one, you're not really evaluating peoples performance. You're just going, well whoever asks gets. So that means like, I either need to be that guy who ask for the raise. Like you know that just not how I feel. You know, like I do my work and I don't necessarily want to ask for a raise. Or like I just need to quit and go to a company that like actually evaluates performance. Um, and then, you know so you could really make the person who doesn't get the raise feel like pretty pissy about it. And don't think that when someone is walking down your company doing the "Schmoney Dance" that other people aren't going to notice.

**Ben Horowitz:** Like they are going to be fired up about that raise. There is - oh you can say this is highly confidential that I am giving you this raise. It's not confidential.

**Ben Horowitz:** Now. And then the cultural conclusion is going to be everybody in your company is going to feel that they now have a fiduciary responsibility to their family to ask for a raise all the time because if they don't then they may be missing out on a raise that they would have otherwise gotten. And like you talk to any experienced CEO and they will tell you this is true. Like if you give out raises when people just ask you for them, like that - you will have a lot of people asking you for raises. That is called encouraging behavior. Um. So what do you do? And really the right answer on raises is you have to be formal. And you have to be formal to save your own culture. Uh and I know this is always this is the thing that causes people running start ups fits because it's like, " Well I don't want like a lot of formalities. I don't want a lot of process. I want it to be organic. We want to do yoga."

**Ben Horowitz:** Um, “We want to only smoke organic weed." Uh, sorry that was like a Peter Theil kind of way. (Ben Horowitz laughs) Peter got very focused on who was smoking weed a little while ago. Um, (Ben Horowitz laughs) but like the process actually protects, um, it protects the culture because what it does it says look we're going to look at all inputs. We are going to have a formal way of saying anybody who wants a raise come talk to me. Like, I'm not going to give you a raise but I am happy to hear your story. Um, I'm going to talk to all the people you work with so I get like a understanding. And I am going to evaluate all the work that you've done, um so I know like where I actually rate and what my
actual opinion on. And I am going to do it periodically. I'm not going to do it daily. But, yeah like
maybe if I were fast moving I would doing it every six months or even maybe once a quarter. Um, and at
the end of that process I will tell you what your raise is. Um, and I will tell you if you're getting one or
if you're not getting one. But I'm not going to do things off cycle. I'm not going to do things when
asked. There is one process and that's it. And like you know when I used to be CEO, and I had like
Executives, its - the bigger you get, the harder this gets because the more aggressive the people
working for you are. Because to get to be an Executive in turns out you have to often be pretty
aggressive, uh, in this world. And in most companies like that's how you get to that level. You know, I
would go, “Look, you can lobby me all you want you know, after the process is done and I give you your
raise but you know what? I am not hearing it. Because I already went through my process. I got your
input, going in. I got everybody else's input. I've got so many people. I've got so much money and you
got like what I believe is right.” And having a process like that basically gets people to be actually more
comfortable. They're more comfortable because they don't have to always be on edge about like, “Am I
asking for what I deserve? Or am I getting like aced out because of who I am, what I look like? I am
not buddy buddy. I'm not at the golf course with you or I'm not doing whatever you like to do? You
know, I don't have to worry about any of that because I know your process. I know your process is like
you're going to evaluate everybody and then you are going to give them what's fair.” And so that's a
much better way to handle that and it means that you're actually understanding what everybody thinks
not just the people you are talking to at the moment.

Ok so now we are going to get into some fun stuff. We are going to evaluate Sam's Blog Post. Um, which
is actually, there are some very good things in it. Um, and there's some things I am going to discuss.

Ben Horowitz: So this is the expert - exert. "Most employees have only have 90 days after they leave a
job to exercise their options. Unfortunately this requires money to cover the strike price and the tax
bill..." And I'll explain this a little more later. Just - but I want to read it first. Uh, “... for the year of
exercise." Blah, blah, blah. "This often more cash than what the employee has." And this is the key.
"And so the employee often has to choose between like leaving the job and walking away from the
vested options i.e. the money that she has because she can't afford to exercise or being like locked into
staying with the company basically for all the wrong reasons. So it's a particularly bad situation when
an Employee gets terminated. And I'll get into that and that's a really key point. "This doesn't seems
fair. The best solution I have I've heard is from Adam D'Angelo, "(a very, very smart guy) "at Quora. The
idea is to grant options are exercisable for 10 years from the grant date, which should nearly all the
cases." Um, whatever happens with the company? Uh, "There's some tricky issues to this." Blah, blah,
blah. Um, “But it's still far better than just losing the assets. I think that this is policy that all startups
should adopt."

So our questions is, like was Sam right? Is this a policy that all startups should adopt? So let me first
explain again what the policy is. So currently, the way almost every stock option package in Silicon
Valley works and in all of Startup world is this - that your stock vest - you get stock invest over a period
of time. Um, but if you leave the company - when you leave you have (and it depends on the company
but think it's 90 days to exercise.) So, yeah - so 90 days and if you do not buy your stock in that
period, it's GONE! Like, it's not yours anymore. Which, depending on when you entered the company
could be a big problem. So a lot of companies today that are valued a lot. Like if you take like a really
valuable Startup say like an Air BnB or an Uber or something like that. When they bring you in they go
“Wow you know like if you look at your 409A price and to the preferred price. Like the stock we're giving
you right now the options are already worth like 10 million dollars." And your like" WOOOH! 10 million
dollars. I'm rich."
Ben Horowitz: Uh, but what they don't necessarily tell you is in order for you to get that money because what the preferred is worth 10 million dollars - Your options are probably going to cost you like 2 1/2 million dollars when you leave and if you don't have that 2 1/2 million dollars in 90 days, it's gone! Like you just lost all your money. Um, so Sam is like, wow! That's fucked up! And so he wrote a blog post.

Ben Horowitz: and he said everybody should change it. So the first question that you have to ask yourself on something like this is "But why is that like - this has kind of been around since like the 80's so why is a rule like this around for 30 years?" And it turned out - Sam, I don't even know - I don't know whether he figured like figured this out or just intuited, but he was right. Like something actually had changed. So up until 2004, there used to be this law called, um, APB Opinion Number 25. That law was the old way to account for stock options and you know it's also the law that all the guys went to jail on. So I got a lot of people who caught a case on APB 25. So I'm glad it's gone. Um, 'cause it's a very confusing law and a lot people did not understand it and they literally went to jail. Uh, but when that was a law, if you gave somebody 10 years to exercise their options, you would never have been able to go public and you would never have been able to be acquired because you basically were taking an expense that was tied to your stock price. So basically the more your stick went up the more compensation expense you'd have to take. And the worse thing about it would be is you wouldn't know what it was going to be. So it would be totally unpredictable. So you could never forecast earnings. Ever! Because your earnings would be a function of all of your stock price. And so the more your stock price went up, the more money you would lose. And in those days people did not look through expense - stock option expenses. So it just wasn't doable. And that's why everybody's agreement was written at 90 days. So that's why it's there. So absolutely it's the right thing to question it being there. Are you guys following? You get this? This is like more complicated than the first two examples but a very important one.

Ok, then - so your perspective on this, if you got Employees is you want to be fair. Like nobody wants a like hire - "Hey you got all this stock in 4 years... SIKE!" Right, and especially like when you fire someone. "Hey you're fired! I feel real bad about it BUT guess what? You know, I am also going to take all your money too!"

Ben Horowitz: Must not feel that bad. So, like that's kind of like a problem. Um, but you also, and this is the thing that you have to keep in mind - you also have to think about the people who are staying and you want to reward the people who are staying. Now the Employee - the perspective of the Employee who leaves - and this is really critical because this is your reputation. Right, like I worked, like you know a year's work - like where's my years pay? Um, ok so now you're telling me about this 90 day exercise and I know it was like in the fine print of my stock option agreement but my Hiring Manager never told me about that. They never told me I was going to need like 2 million dollars to get my stock. Which I don't have. Um, so meant like so if I was rich like I'd get my stock? Like that's not fair. Uh, so like now I am fired and then I'm screwed. And guess what? You know I am going to tell EVERYBODY like how you like screwed me over. And so that's a real reputational problem. So that's something that you consider as and saying as policy. But then you also have to consider the Employee who stays. And, one thing that they're going to ask themselves is look they're leaving and every time anybody leaves it's like - was that smart? Like that's something that you - Because these are people who right - your Employees know each other better than they know you. In any company I don't care what company you are. But like, often the like the person they're really working with is it going to be the person they know more? And so if that person leaves they're going to go " Well, should of I left too? Like what did they get and how does that compare to my deal?"
And so if we look at the situation, and we try and analyze it, there's a lot of components to it. So first it's like, companies uh, tread a lot of people around here and I think like really the average is like somewhere around 10 percent. It's probably getting higher particularly if you are in San Francisco it's getting higher. And just 'cause of the nature of the culture there. Um, and then Silicon Valley companies dilute like 6-8 or even 10 percent a year for Employee options. And you have to keep in mind that as mean as it maybe, if that Employee leaves and can't exercise their options, um, then those options come back to the pool where you can potentially give them to people who are already there. So, you're actually taking less dilution. Um, so that's something that you have to think about. I am not saying you have to act on it but it's something that you have to think about it.

Um, then secondly - Look losing all your stock is a very big incentive to stay. And that could be good news or bad news. Right? Like it could be good news in that you get to keep somebody you might have lost. And it could be bad news in that you kept them for the exact wrong reason. Right, 'cause they have handcuffs on them. And so you know you may get like Exec - the Employee that is worse than not having an Employee. Um, but on the other hand - a 10 year option on a highly volatile security - for those of you who have taken that class. Anybody taken that class? That's valuable. Right, 10 years option are - volatility and length. That's the value of an option. Well 10 years on a Startup stock, that's a big valuable thing. And then remember the employee stays doesn't get that. The Employee stays just gets a stock. They get but they don't get the new job and the new stock. So they get one thing but they don't get both things. And so you've got to weigh that in. So this is a hard one. Like I think that it should be reevaluated by every company. Um, I wouldn't go as far as to Sam as it should be adopted by every company. I think you have to think about what you want. And I would just offer kind of two Alternative Cultural Statements. One is, look - we treat employees with the up straightest forwardness. We're going to be fair and therefore you get like 10 years to exercise your stock. And like what we said we're going to give you, you're gonna get regardless of how rich or poor you are. Like that's just a done deal.

Um, the second way to handle it, which is - and no company companies do this. Which is why I actually really like this post that he wrote. Look, you can say up front " Look you are guaranteed to get your salary but for your stock to be meaningful, like these are the things that have to happened. One you've got to vest. Two you have to stay until we get to an exit. Like company makes it. And/ or like you've got other money. And like finally, like the company actually got to be worth something. 'Cause 10 percent of nothing is nothing. Um, and look, the reason we set the policy this way is like we really value people who stay. So don't join this company, like if you are going to join another one in 18 months 'cause like you're going to get screwed. And like our policy like guarantees you're going to get screwed on that.

And so, like those are two ways to handle it. It really depends on like you and like how you want to run your culture. But again with all these things, it's just critical to think it through from everybody's prospective because when push comes to shove that's gonna matter. That's going to change the outcome of your company.

Sam: I am actually revising my recommendation slightly.

Ben Horowitz: (Chuckles) let's hear it.

Sam: No, it's that I think there needs to be more incentive to stay. Through - But still if someone gets fired, I still think they get screwed a lot of the time.
Ben Horowitz: Well the other and the other thing that's really important, um that Sam pointed out - the distinction now is how much money you have. Right, if you have the money, you don't get screwed. You walk away with all - you can buy your stock. Um, you do take some risks - but you can buy your stock. If you don’t have the money - you don't have the money.

Ok. So now we're getting to the person on my shirt Toussaint. He was the best at this and I want to take you through some examples because they're very powerful. Ok. So first of all about Toussaint. He was born - the thing to understand about him - he was born a slave. And um he wasn't just born a slave - he was born a slave um in the most brutal place to be a slave, uh which was uh the Colony of Santa Domingo you know now referred to as Haiti. Um, but this was actually a much more severe form of slavery as were kind of all the sugar growing areas then even US slavery which is historically a very brutal form of slavery. And just to give you some numbers on it, um basically over the course of slavery the - somewhere like 400 years, um a million slaves were brought to the US and at the end of slavery, there were four million slaves in the US. In that same period to the sugar growing countries in the Caribbean, two million slaves were brought over and at the end of slavery there were seven hundred thousand left. So from just a quantitative perspective, like nearly 10 times more brutal. Um, and um, I am going to read this to you. I don't know if I quite have time but I don't care.

Ben Horowitz: But just to give you an idea, this is just like a quantitative thing. Um, I'll read you sort of a description of slavery in Toussaint's area. "Whipping was interrupted in order to pass a piece of hot wood on the buttocks of the victim. Salt, pepper, citrus, cinders, aloes and hot ashes were poured into bleeding wombs. It's not to heal them. This is to make it worse. Mutilations were common. Limbs, ears and sometimes private parts to deprive them of the pleasures which they could indulge without expense. Their masters poured burning wax on their arms and hands and shoulders. Emptied the boiling king sugar over their heads. Burning them alive. Roasting them on the slow fires. Filled them with gun powder and blew them up with a match. Buried them up to their neck and smeared their head with sugar that that the flies might devour them. Fastened them to the nest of ants or wasps. And made them eat the excrement, drink their urine, lick saliva of other slaves. One Colonist was known "in moments of anger to throw himself on a slaves and stick his teeth into their flesh."

So that's the slavery that he grew up in. Um, and it's really important to understand this because to get out of that perspective um, was not easy. Um, but he had a vision. And his vision was kind of threefold. One he wanted to end slavery. Two he wanted to actually take control of the country and run the country. And thirdly, he wanted it to be a first class world class country. Not just like something that where he had freed the slaves but something that could compete on a worldwide basis. Uh so that was his mindset going in but that was the background that he came from.

So example - management example - One conquering the enemy. So, uh the kind of sequence of battles that occurred um in Haiti were first - you know he had to kind of defeat the locals. Uh, but then once he defeated the locals which is there were several countries that were very, very interested in taking control of Haiti. Principally Spain, England and France. Uh, so he had to defeat those armies as well. When he conquered them; um, he had to decide what to do with the kind of conquered soldiers and the leaders on the other side. And to this he really took into perspective, uh kind of three different points of view. One, his soldier’s point of view. Two his enemies point of view. And finally the point of view of the resulting culture. Like what kind of country was he building because the army was going to be the seed corn for the culture of the whole country?

Um, so from the soldier’s perspective - I get this - do we get to pillage? Like soldiers like to pillage. Um, they get stuff. It's something for their work. Um, and like the second thing is they're trying to kill us so we should kill them. So that's a basic perspective. Of the people who are fighting for him. So the most
important people to Toussaint. Now, I put pillage up there in - so a couple of things to know - One I didn't put rape up there. And very interestingly like not only did he not allow rape among his army; but he didn't even allow his officers to cheat on their wives. And if they did he would get rid of them because he was so concerned about the resulting culture. What was it going to be? Was it going to be productive? Was it going best in world? Or was it going to be something less than that? And so that was his mindset going in. His army was actually famous for not pillaging. So they were already actually used to this. They were famous for not pillaging and this one of the most surprising things to the conquered people to the point where, um in Haiti he had a reputation where even the white people were like very impressed just because he would go in and go into their city and not pillage even though he would win. Um again, this is because he took a long view of the culture.

So, and this is kind of an important subtle point um which gets him to his conclusion. But he believe the culture of Haiti because it was a slave culture, sugar plantation culture was just pretty low grade as to what he had experienced in Europe when he dealt with the Europeans. And then he thought that slave culture was even more broken than Haitian culture because if you think about slave culture it's like the kind of culture where, oh you don't do what I tell you I'm going to beat you to death. I am going to blow up with gun powder. If you think that kind of behavior that that, then that ensues from that - that was the culture he knew he needed to replace. He knew he needed to upgrade. So his solution was when he conquered the British. Or he conquered the Spanish. Or he conquered the French - he would take the very best people from there and he would make them generals in his army. Like, so uh - You probably didn't expect that. (Chuckles) So here are the guys trying to kill him. He leading a slave evolution and he - when he conquers the enemy he actually incorporates them into his army and makes them part of that because he wanted the expertise and he wanted the - he wanted the um, the culture to be at a much higher level.

So the second question he had - this even more complicated. What do you do with the slave owners? So you're leading slave revolution, you take control of the country - what do you do with the slave owners? Three perspectives again. Um, so for the slaves - like come on - like if you're a slave and you win the war against the slave owners like you want to kill them. There is no question. And not only do you want to kill them - like that's your land now. Like we won. Like F you. Um, from Toussaint's perspective it was more complicated because he wanted Haiti to be a first world country and sugar was really important. And the whole slave economy was the sugar economy. Then, right like on the other hand he was a slave and he's got to be pretty upset. Particularly given the type of slavery. Um, then, uh but he had to consider like, he didn't know how to run a sugar plantation. Um and then, like and he didn't have like any business relationships on who to trade the sugar with. Um, but on the other hand like, was. You win the war you get the land. Like that's a pretty basic rule. So what to do? Um, and then if you look at the slave owner perspective. It's pretty interesting because they're coming at it - and this is the point of view that he actually had the discipline to understand - They were coming from a cost structure that was predicated on slave labor. So like, their business didn't work without slave labor. Like literally if they had to pay people like their cash flow wouldn't work. Um, they paid a lot of money for the slaves up front. And they paid a lot of money for the land. So in their mind like, to run it, like that was like that's how business works. Like you can just, you can't just like change the economics and have it still work. Um, and then they knew they had like some power because of the position they were in.

Um, so what was the answer for the slave owners? So, one so the solutions was one - I'm going to end slavery. Two - I am going to let the slave owner keep their land. Three - I am going to make them pay their workers. So there are no more slave labor. You have to have paid workers. But in order to fund that I am going to lower their taxes. (Paused) You guys ought to be kind of impressed with that.
Ben Horowitz: Like lower the taxes of the slave owners after you defeat the slave owners and like end slavery. Um, but he bigger goal. He wanted a stronger culture. The way he treated those slave owners, the need to keep the economy going was important. And then like let's look at the results. So first of all, it is Toussaint's revolution is the successful slave revolution in the history of mankind. There has never been another one and there may never hopefully we won't have slavery in a big way there won't be another one so like he's it. Um, two, um you know the plantation owners kept their land. Three he defeated Neapolitan. He had a booming economy and a world class culture. Under Toussaint - Haiti had more exports - export revenue than the United States. So that's how successful he was in the revolution. Um, and this is the power of looking at situation not just from your point of view - but from the point of view of all the constituents. Even the people you hate. Which is hard to do when you are CEO and harder to do when you are leading the revolution.

Um, so just look in conclusion, the most important think that you can learn and see one of the hardest things to do is you have to discipline yourself to see your company through the eyes of the people you are working through. Through the eyes of the employees. Through the eyes of your partners. Through the eyes of the people you are not talking to and who are not in the room. Thank you.

Now I will take questions.

Audience member 1: 37:08 ...

Ben Horowitz: Right, this is great question. Yes, so the question is, if you got to fire or demote an Executive, one how do you have the conversation and and then two how do you explain it to everyone else? 'Cause like it's clearly some kind of failure. You failed on hiring. You failed on integrating. They failed at their job. Like, some - it's a failure. Um, and so - alright say look - the first thing is um, when firing the person you, have to really try to be honest. Um, and you know you're feeling like you failed. And I think a common reaction is like - then there is a couple of common reactions - One, is like you just suck. And like so like I am firing you. Screw off. That's not good 'cause it's not really true. Um, it may be like you're feeling that way. And another kind of common mistake is just to be like too mushy. It's not you it's me. And like it feels like some kind of weird break up. Um, with an ex-boyfriend that you really didn't like. But generally look, when you hire people, look you try to hire the very best. And you hire people who are qualified to do the job. And generally the reason they fail on the job is you made some mistake in the hiring process and that you didn't match them to the needs of your company accurately enough. That's the number one reason why this fails and so that's generally a good place to start. To say look - here's how we are and here's what I didn't recognize about us and about you when I made the decision and now like it is what it is. Um, so we're going to have to move on. Uh, and then when you talk to the Employees about it. Like this - and this gets different. Which is - Look you can take somebody's job - you have to take their job. And this is something Bill Campbell taught - I but you don't to take their dignity. And so it's not necessary to get up in front of the company and I blew that muther fucker out. I capped his ass.

Ben Horowitz: Um, in fact it's not good. 'Cause nobody feels good about that. Like you know, you might feel like proud of yourself but nobody else feels good about that, you know the right thing to do is just look - Thank them for their work. Like let people know that their moving on. Um, and you don't really have to explain all their personal details. It's more important to leave them with their dignity. Uh, and let them go on to live another day because look, what you say at that meeting that's their reputation because everybody in your company is going call on that person when they try and get their next job. So if you start saying like a bunch of BS about them - Like that's not going to be good and it's not going get interpreted as like we screwed up - it's going get interpreted he screwed up. And so there
are kind of two different things. You have to be very honest with them but you have to make sure you preserve their dignity when you talk to the company.

Audience member 2: So um, I was reading your book yesterday and um, how did you in particular dealt with all the stress? Was it like, was it like meditating? Hip Hop?

Audience member 2: 40:36

Ben Horowitz: (Laughs) so the question was - How do I deal with all the stress of being CEO? And uh the answer is I used to be 6 foot 4 and good looking.

Ben Horowitz: So clearly not very well. Umm, I get asked that a lot and I really have a great answer for it. I mean I think that the um, one I have a wonderful wife who is sitting right here. So I will oar and he borrowed that technique from him but applied it in a much kind of more dramatic context. Uh, and uh so his army was (chuckles) kind of - he had British, um French - um Spanish - slaves and mulattoes who most of the mulattoes in Haiti at that time were pro slavery. Um, so that was another issue but his leadership was so great - everybody wanted to join him.

Audience member 4: How do you incorporate that - same ideology to people who were previously against you on your side? Uh, and I am sorry the question was - How do you get people - How do you incorporate Toussaint's ideology and get people who were previously against you on your side? And uh, I would just say like what he did in general is the right thing is to just basically you have to show them a better way. Um, you know as leader - if somebody's your enemy and you need to convert them over and this happens in business too where somebody is like a competitor something and you want to bring them over but you want to bring them over you know don't want to bring all the in ethnically people who switch competitor to competitor over. And it really is your culture has to be elevated your mission has to be elevated. Your way of doing things has to be just better. Uh, and that was the thing that was so compelling for the other rest of the army.

Audience member 5: So going off of your last I am curious to sort of like interested in that you founded or co-founder. How have you built like a culture around people and along the entrepreneurs that you work with that has sort of differentiated you in the market from all the other?

Ben Horowitz: Um. Probably not the best question for me. I can ask Sam that. I don't know. Uh, so the question is how do we build a culture out of that is differentiated us from all other VC's. I feel like, um, that is certainly the goal and we have been around certainly for 5 years now and uh, you know the attempt that we made at it um and uh it's for the rest of the world to judge if we succeeded was basically this - um you know in the old days of VC (chuckles) when I was a entrepreneur, the basic idea was you have an entrepreneur or inventor and they get a company to a point and at that point they either be ready to be CEO or you would go find a CEO to replace them and build quote on quote the company. Uh, our kind of cultural philosophy was like look - the founder and inventor was special. So we're going to design the firm and the culture of the firm to help the founder develop into a CEO and so we do a lot of systematic things different. Um, the two biggest are one - all of our partners are kind of founders - CEO's. Um so that you get somebody - original model with some sort of experience required. Um - that's a joke.

Ben Horowitz: If you are advised a CEO you got to kind have been a CEO. Imagine that. That's why I like Sam. He used to be a CEO. He doesn't talk about it that much but he was CEO and good at it. Um, and then uh, the second part is that a professional CEO will bring in - in the old days - is a network just
tons of people that he would know from like all kinds of you know guys who brought technology in big
corporations to like important partners in the field like guys on Google and Facebook to uh, you know
people in the press that he or she might know and so forth. And so we try to build that network on your
behalf uh, at the firm. And I think we do a better job of that than anybody else. So those are the way
that we try to be different.

Sam: Uh one more question...

Ben Horowitz: One more question yes....

Audience member 6: Is it like, um, putting yourself in other people's shoes is very important I can
imagine that. So can you give us some tips to do it 46:52?

Ben Horowitz: Yeah so, so putting yourself in other people's shoes is difficult in management. Can you
think about how you can do it in daily life? It's hard in daily life. It's even harder in management
because it's the stress at the moment, right? Like a great Employee is asking you for a raise it is very
hard for not to respond. It's, it's a - 'cause you do not want to lose them and they are not asking you for
a raise randomly they are asking you a raise for a reason. Um, so to say ok, if you don't have a process
in place to go stop back out - uh I would just say - and it's a key thing in being a leader is you got to
pause yourself. Like if somebody comes to you with something that you know is important um, but you
want to feel like - you know when you're a leader you want want to feel like you have all the answers.
Like right now you guys are asking me questions if I don't know the answer I will make something up.

Ben Horowitz: Because I want you guys to think I am smart. Um, and it is important - Like I said the
most important thing is to pause. Like if you know something is really important and you haven't though
it through just to say you know look - I am taking this really seriously but I have to pause. Because I
have to think it through from all perspectives. And I'm going to come back. And I end up doing that a
lot just because there are a lot of things that you run into that you haven't never seen before. And I tell
you, most CEO's, including myself learn this the hard way, you walk in um you kind of step in like three
or four times. You go ok, I'm going sneak away with this. Nobody's going to see me give them the raise.
And I'm going to do it and it's gonna be all under the covers. Confidentiality baby. Uh and then it comes
up and blows up in your face three weeks later and you're like, Oh my God. Like what have I done? Or
three months later or even a year later. Um and then once it's a year later it's a huge problem. Like so
you've take what's a little emotional problem and you've turned it into a forest fire. Uh, we call it a
Kenichi problem. The deeper you bury it the hotter it gets. Um, you know it's a Korean joke.

Ben Horowitz: But uh (laughs) But you know, it takes practice. Uh I say it takes practice. It's very
difficult to do. And I'd say, you know like some of the - My friend Bill Campbell. This is his big skill like
this what he is so great at people always trying to describe him to me and I'm like that is not him at
all. That's not what he's good at. Like he's not good at that or that but he's good at this. He's good at
seeing the company through the eyes of the employees and um like I said if you are good at that you
will be in a very likely an elite leader.

So thank you. Thank you
The shit sandwich can work well with junior employees, but has the following challenges:

- It tends to be overly formal. Because you have to preplan and script the sandwich to make it come out correctly, the process can feel formal and judgmental to the employee.
- After you do it a couple of times, it will lack authenticity. The employee will think: “Oh boy, she’s complimenting me again. I know what’s coming next, the shit.”
- More senior executives will recognize the shit sandwich immediately and it will have an instant negative effect.

Early in my career, I attempted to deliver a carefully crafted shit sandwich to a senior employee and she looked at me like I was a little kid and said: “Spare me the compliment, Ben, and just tell me what I did wrong.” At that point, I thought that I was definitely not born to be a CEO.

The Keys

To become elite at giving feedback, you must elevate yourself beyond a basic technique like the shit sandwich. You must develop a style that matches your own personality and values. Here are the keys to being effective:

- **Be authentic.** It’s extremely important that you believe in the feedback that you give and not say anything to manipulate the recipient’s feelings. You can’t fake the funk.
- **Come from the right place.** It’s important that you give people feedback because you want them to succeed and not because you want them to fail. If you really want someone to succeed, then make her feel it. Make her feel you. If she feels you and you are in her corner, then she will listen to you.
- **Don’t get personal.** If you decide to fire somebody, fire her. Don’t prepare her to get fired. Prepare her to succeed. If she doesn’t take the feedback, then that’s a different conversation.
- **Don’t clown people in front of their peers.** While it’s OK to give certain kinds of feedback in a group setting, you should strive never to embarrass
Lecture 15: How to Manage

someone in front of their peers. If you do so, then your feedback will have little impact other than to a) cause the employee to be horribly ashamed and b) cause the employee to hate your guts.

- Feedback is not one size fits all. Everybody is different. Some employees are extremely sensitive to feedback while others have particularly thick skin and often thick skulls. Stylistically, your tone should match the employee’s personality not your mood.
- Be direct, but not mean. Don’t be obtuse. If you think somebody’s presentation sucks, don’t say: “It’s really good, but could use one more pass to tighten up the conclusion.” While it may seem harsh, it’s much better to say: “I couldn’t follow it and I didn’t understand your point and here are the reasons why.” Watered down feedback can be worse than no feedback at all because it’s deceptive and confusing to the recipient. But don’t beat them up or attempt to show your superiority. Doing so will defeat your purpose because when done properly feedback is dialogue, not a monologue.

Feedback is a dialogue, not a monologue

You may be the CEO and you may be telling somebody about something that you don’t like or disagree with, but that doesn’t mean that you’re right. Your employee should know more about her function than you. She should have more data than you. You may be wrong.

As a result, your goal should be for your feedback to open up rather than close down discussion. Encourage people to challenge your judgment and argue the point to conclusion. Culturally, you want super high standards thoroughly discussed. You want to apply tremendous pressure to get the highest quality thinking, yet be open enough to find out when you are wrong.

High frequency feedback

Once you’ve mastered the keys, you should practice what you’ve mastered all the time. As CEO, you should have an opinion on absolutely everything. You should have an opinion on every forecast, every product plan, every presentation and even every comment. Let people know what you think. If you like someone’s comment, give her the feedback. If you disagree, give her the feedback. Say what you think. Express yourself.

This will have two critically important positive effects:

- Feedback won’t be personal in your company. If the CEO constantly gives feedback, then everyone she interacts with will just get used to it. Nobody will think: “Gee, what did she really mean by that comment? Does she not like me?” Everybody will naturally focus on the issues, not an implicit random performance evaluation.

- People will become comfortable discussing bad news. If people get comfortable talking about what each other are doing wrong, then it will be very easy to talk about what the company is doing wrong. High quality company cultures get their cue from data networking routing protocols: bad news travels fast and good news travels slowly. Low quality company cultures take on the personality of the Wicked Witch of the East in The Wiz: “Don’t nobody bring me no bad news.”

Making the CEO

Being CEO requires also a broad set of more advanced skills—I’ve written about many in this blog—but the key to reaching the advanced level and feeling like you were born to be CEO is mastering the unnatural.

If you are a founder CEO and you feel awkward or incompetent when doing some of these things and believe there is no way that you’ll be able to do it when your company is 100 or 1,000 people, welcome to the club. That’s exactly how I felt. So did every CEO that I’ve ever met. This is the process. This is how you get made.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Lecture 15: How to Manage

Ben Horowitz

At Opsware I used to teach a management expectations course because I deeply believed in training. In it, I made it clear that I expected every manager to meet with her people on a regular basis. I even gave instructions on how to conduct a 1:1 meeting so there could be no excuses.

Then one day while I happily went about my job, it came to my attention that one of my managers hadn’t had a 1:1 with any of his employees in over six months. While I knew to “expect what I inspect,” I did not expect this. No 1:1 in over six months? How was it possible for me to invest so much time thinking about management, preparing materials and personally training my managers and then get no 1:1s for six months? Wow, so much for CEO authority. If that’s how the managers listen to me, then why do I even bother coming to work?

I thought that leading by example would be the sure way to get the company to do what I wanted. Lord knows the company picked up all of my bad habits, so why didn’t they pick up my good habits? Had I lost the team? I recalled a conversation I’d had with my father many years ago regarding Tommy Heinsohn, the Boston Celtics basketball coach at the time. Heinsohn had been one of the most successful coaches in the world, including being named “coach of the year” and winning two NBA championships. However, he had gone downhill fast and now had the worst record in the league. I asked my father what happened. He said: “The players stopped paying attention to his temper tantrums. Heinsohn used to yell at the team and they’d respond. Now they just ignore him.” Was the team now ignoring me? Had I yelled at them one time too many?

The more that I thought about it, the more I realized that while I had told the team “what” to do, I had not been clear about “why” I wanted them to do it. Clearly, my authority alone was not enough to get them to do what I wanted. Given the large number of things that we were trying to accomplish, managers couldn’t get to everything and came up with their own priorities. Apparently, this manager didn’t think that meeting with his people was all that important and I hadn’t explained to him why it was so important.

So why did I force every manager through management training? Why did I demand that managers have 1:1s with employees? After much deliberation with myself, I settled on an articulation of the core reason and I called up the offending manager’s boss—I’ll call him Steve—and I told him that I needed to see him right away.

When Steve came into my office I asked him a question: “Steve, do you know why I came to work today?”

Steve: “What do you mean, Ben?”

Me: “Why did I bother waking up? Why did I bother coming in? If it was about the money, couldn’t I sell the company tomorrow and have more money than I ever wanted? I don’t want to be famous, in fact just the opposite.”

Steve: “I guess.”

Me: “Well, then why did I come to work.”

Steve: “I don’t know.”

Me: “Well, let me explain. I came to work, because it’s personally very important to me that Opsware be a good company. It’s important to me that the people who spend 12 to 16 hours/day here, which is most of their waking life, have a good life. It’s why I come to work.”

Steve: “OK.”

Me: “Do you know the difference between a good place to work and a bad place to work?”

Steve: “Umm, I think so.”

Me: “What is the difference?”

Steve: “Umm, well…”

Me: “Let me break it down for you. In good organizations, people can focus on their work and have confidence that if they get their work done, good things will happen for both the company and them personally. It is a true pleasure to work in an organization such as this. Every person can wake up knowing that the work they do will be efficient, effective and make a difference both for the organization and themselves. These things make their jobs both motivating and fulfilling.

“In a poor organization, on the other hand, people spend much of their time fighting organizational boundaries, infighting and broken processes. They are not even clear on what their jobs are, so there is no way to know if they are getting the job done or not. In the miracle case that they work ridiculous hours and get the job done, they have no idea what it means for the company or their careers. To make it all much worse and rub salt in the wound, when they finally work up the courage to tell management how fucked up their situation is, management denies there is a problem, then defends the status quo, then ignores the problem.”

Steve: “OK.”

Me: “Are you aware that your manager Tim has not met with any of his employees in the past six months?”

Steve: “No.”
Me: “Now that you are aware, do you realize that there is no possible way for him to even be informed as to whether or not his organization is good or bad?”

Steve: “Yes.”

Me: “In summary, you and Tim are preventing me from achieving my one and only goal. You have become a barrier blocking me from achieving my most important goal. As a result, if Tim doesn’t meet with each one of his employees in the next 24 hours, I will have no choice but to fire him and to fire you. Are we clear?”

Steve: “Crystal.”

Was that really necessary?

You might argue that no matter how well managed a company is, it will fail without product/market fit. You might argue further that horribly managed companies that achieve massive product/market fit succeed just fine. And you would be right on both accounts. So was it really necessary for me to make such a dramatic speech and threaten one of my executives?

I think it was for the following three reasons:

• Being a good company doesn’t matter when things go well, but it can be the difference between life and death when things go wrong.
• Things always go wrong.
• Being a good company is itself an end.

The difference between life and death

When things go well, the reasons to stay at company are many:

• Your career path is wide open because as the company grows lots of interesting jobs naturally open up.
• Your friends and family think you are a genius for choosing to work at the “it” company before anyone else knew it was “it”.
• Your resume gets stronger by working at a blue chip company in its heyday.
• Oh, and, you are getting rich.

When things go poorly, all those reasons become reasons to leave. In fact, the only thing that keeps an employee at a company when things go horribly wrong—other than needing a job, which isn’t so applicable in the current macro environment—is that she likes her job.

Things always go wrong

There has never been a company in the history of the world that had a monotonically increasing stock price. In bad companies, when the economics disappear, so do the employees. In technology companies, when the employees disappear, the spiral begins: the company declines in value, the best employees leave. Spirals are extremely difficult to reverse.

Being a good company is an end itself

When I first met my friend Bill Campbell, he was chairman of Intuit, on the board of Apple and a mentor to many of the top CEOs in the industry, including Steve Jobs and Jeff Bezos. However, those things did not impress me nearly as much as his time running a company called GO Corporation. GO essentially attempted to build an iPhone in 1992. The company raised more money than almost any other venture capital back startup in history and lost nearly all of it before selling itself for nearly nothing to AT&T in 1994.

Now that probably doesn’t sound impressive. In fact, it probably sounds like a horrible failure. But I’d met tens of GO employees in my career, including great people like Mike Homer, Danny Shader, Frank Chen and Stratton Sclavos, and the amazing thing was that every GO employee that I’d ever met counted GO as one of the greatest work experiences of their lives. The best work experience ever despite the fact that their careers stood still, they made no money and they were front-page failures. GO was a good place to work.

This made me realize what an amazingly effective CEO Bill was. Apparently, John Doerr thought that too, because when Scott Cook needed a CEO for Intuit, John recommended Bill even though Bill lost a ton of John’s money at GO. And for years, everyone who ever came into contact with GO employees knew what Bill was about. He was about building good companies.

If you do nothing else, be like Bill and build a good company.
Lecture 15: How to Minimize Politics in Your Company

http://www.bhorowitz.com/how_to_minimize_politics_in_your_company

Ben Horowitz

In all my years in business, I have yet to hear someone say: “I love corporate politics.” On the other hand, I meet plenty of people who complain bitterly about corporate politics—sometimes even in the companies they run. So, if nobody loves politics, why all the politics?

Political behavior almost always starts with the CEO. Now you may be thinking: “I hate politics, I’m not political, but my organization is very political. I clearly didn’t cause this.” Sadly, you needn’t be political to create extreme political behavior in your organization. In fact, it’s often the least political CEOs who run the most ferociously political organizations. Apolitical CEOs frequently accidently encourage intense political behavior.

What do I mean by politics? I mean people advancing their careers or agendas by means other than merit and contribution. There may be other types of politics, but politics of this form seem to be the ones that really bother people.

How it happens

A CEO creates politics by encouraging and sometimes incenting political behavior—often accidentally. For a very simple example, let’s consider executive compensation. As CEO, senior employees will come to you from time to time and ask for an increase in compensation. They may suggest that you are paying them far less than their current market value. They may even have a competitive offer in hand. Faced with this confrontation, if the request is reasonable, you might investigate the situation. You might even give the employee a raise. This may sound innocent, but you have just created a strong incentive for political behavior.

Specifically, you will be rewarding behavior that has nothing to do with advancing your business. The employee will earn a raise by asking for one rather than you automatically rewarding them for outstanding performance. Why is this bad? Let me count the ways:

1. The other ambitious members of your staff will immediately agitate for raises as well. Note that neither this campaign nor the prior one need be correlated with actual performance. You will now spend time dealing with the political issues rather than actual performance issues. Importantly, if you have a competent board, you will not be able to give them all out-of-cycle raises, so your company executive raises will occur on a first-come, first-serve basis.

2. The less aggressive (but perhaps more competent) members of your team will be denied off-cycle raises simply by being apolitical.

3. The object lesson for your staff and the company will be the squeaky wheel gets the grease and the political employee gets the raise. Get ready for a whole lot of squeaky wheels.

Now let’s move on to a more complicated example. Your CFO comes to you and says that he wants to continue developing as a manager. He says that he would like to eventually become a COO and would like to know what skills he must demonstrate in order to earn that position in your company. Being a positive leader, you would like to encourage him to pursue his dream. You tell him that you think that he’d be a fine COO some day and that he should work to develop a few more skills. In addition, you tell him that he’ll need to be a strong enough leader, such that other executives in the company will want to work for him. A week later, one of your other executives comes to you in a panic. She says that the CFO just asked her if she’d work for him. She says that he said that you are grooming him to be the COO and that’s his final step. Did that just happen? Welcome to the big time.

How to minimize politics

Professionals vs. Amateurs

Minimizing politics often feels totally unnatural. It’s counter to excellent management practices such as being open minded and encouraging employee development.

The difference between managing executives and managing more junior employees can be thought of as the difference between being in a fight with someone with no training and being in a ring with a professional boxer. If you are in a fight with a regular person, then you can do natural things and they won’t get you into much trouble. For example, if you want to take a step backwards, you can pick your front foot up first. If you do this against a professional boxer, you will get your block knocked off. Professional boxers train for years to take advantage of small errors in technique. Lifting your front foot first to take a step backwards will take you slightly off balance for a split second and that’s all your opponent will need.

Similarly, if you manage a junior employee and they ask you about their career development, you can say what comes naturally and generally get away with it. As we saw above, things change when you deal with highly ambitious, seasoned professionals. In order to keep from getting knocked out by corporate politics, you need to refine your technique.

The Technique

As I developed as a CEO, I found three key techniques to be extremely useful in minimizing politics.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208) 329 / 412
1. Hire people with the right kind of ambition—The cases that I described above might involve people who are ambitious, but not necessarily inherently political. All cases are not like this. The surest way to turn your company into the political equivalent of the US Senate is to hire people with the wrong kind of ambition. As defined by Andy Grove, the right kind of ambition is ambition for the company’s success with the executive’s own success only coming as a by-product of the company’s victory. The wrong kind of ambition is ambition for the executive’s personal success regardless of the company’s outcome.

2. Build strict processes for potentially political issues and do not deviate—Certain activities attract political behavior. These activities include:
   - Performance evaluation and compensation
   - Organizational design and territory
   - Promotions

Let’s examine each case and how you might build and execute a process that insulates the company from bad behavior and politically motivated outcomes.

Performance and compensation—Often companies defer putting performance management and compensation processes in place. This doesn’t mean that they don’t evaluate employees or give pay raises; it just means they do so in an ad hoc manner that’s highly vulnerable to political machinations. By conducting well-structured, regular performance and compensation reviews, you will ensure that pay and stock increases are as fair as possible. This is especially important for executive compensation as doing so will also serve to minimize politics. In the example above, the CEO should have had an airtight performance and compensation policy and simply told the executive that his compensation would be evaluated with everyone else’s. Ideally, the executive compensation process should involve the board of directors. This will a) help ensure good governance and b) make exceptions even more difficult.

Organizational design and territory—if you manage ambitious people, from time to time, they will want to expand their scope of responsibility. In the example above, the CFO wanted to become the COO. In other situations, the head of marketing might want to run sales and marketing or the head of engineering may want to run engineering and product management. When someone raises an issue like this with you, you must be very careful about what you say, because everything that you say can be turned into political cannon fodder. Generally, it’s best to say nothing at all. At most, you might ask “why?”, but if you do so be sure not to react to the reasons. If you indicate what you are thinking, that information will leak, rumors will spread and you plant the seeds for all kinds of unproductive discussions. You should evaluate your organizational design on a regular basis and gather the information that you need to decide without tipping people to what you plan to do. Once you decide, you should immediately execute the re-org: don’t leave time for leaks and lobbying.

Promotions—Every time your company gives someone a promotion, everyone else at that person’s level evaluates the promotion and judges whether merit or political favors yielded the promotion. If the latter, then the other employees generally react in one of three ways:
1. They sulk and feel undervalued
2. They outwardly disagree, campaign against the person, and undermine them in their new position
3. They attempt to copy the political behavior that generated the unwarranted promotion

Clearly, you don’t want any of these behaviors in your company. Therefore, you must have a formal, visible, defensible promotion process that governs every employee promotion. Often this process must be different for people on your own staff (the general process may involve various managers who are familiar with the employee’s work, the executive process should include the board of directors). The purpose of the process is twofold. First, it will give the organization confidence that the company at least attempted to base the promotion on merit and second, the result of the process will be the information necessary for your team to explain the promotion decisions that you made.

3. Be careful with “he said, she said”—Once your organization grows to a significant size, members of your team will, from time to time, complain about each other. Sometimes this criticism will be extremely aggressive. Be very careful about how you listen and the message that it sends. Simply by hearing them out without defending the employee in question, you will send the message that you agree. If people in the company think that you agree that one of your executives is less than stellar, that information will spread quickly and without qualification. As a result, people will stop listening to the executive in question and they will soon become ineffective.

There are two distinct types of complaints that you will receive:
1. Complaints about an executive’s behavior
2. Complaints about an executive’s competency or performance

Generally, the best way to handle complaints of type 1 is to get the complaining executive and the targeted executive in the room together and have them explain themselves. Usually, simply having this meeting will resolve the conflict and correct the behavior (if it was actually broken). Do not attempt to address behavioral issues without both executives in the room. Doing so will invite manipulation and politics.

Complaints of type 2 are both more rare and more complex. If one of your executives summons the courage to complain about the competency of one of their peers, then there is a
good chance that either the complainer or the targeted executive has a major problem. If you receive a type 2 complaint, you will generally have one of two reactions: a) they will be telling you something that you already know or b) they'll be telling you shocking news.

If they are telling you something that you already know, then the big news is that you have let the situation go too far. Whatever your reasons for attempting to rehabilitate the wayward executive, you have taken too long and now your organization has turned on the executive in question. You must resolve the situation quickly. Almost always, this means firing the executive. While I've seen executives improve their performance and skill sets, I've never seen one lose the support of the organization then regain it.

On the other hand, if the complaint is new news, then you must immediately stop the conversation and make clear to the complaining executive that you in no way agree with their assessment. You do not want to cripple the other executive before you re-evaluate their performance. You do not want the complaint to become a self-fulfilling prophecy. Once you've shut down the conversation, you must quickly re-assess the employee in question. If you find that they are doing an excellent job, then you must figure out the complaining executive's motivations and resolve them. Do not let an accusation of this magnitude fester. If you find that the employee is doing a poor job, there will be time to go back and get the complaining employee's input, but you should be on a track to remove the poor performer at that point.

**Closing Thought**

As CEO, you must consider the systemic incentives that result from your words and actions. While it may feel good in the moment to be open, responsive and action oriented, be careful not to encourage all the wrong things.
16. How to run a user interview
Lecture 16: How to Run a User Interview

http://startupclass.samaltman.com/courses/lec16/

Emmett Shear

Sam Altman: Good afternoon. Today's guest speaker is Emmett Shear. Emmett is the CEO of Twitch, which was acquired by Amazon, where he now works. Emmett is going to talk about how to do great user interviews; this is the talking to users part of "How to Start a Start Up". It should be really useful. Thank you very much for coming!

Emmett Shear: Thanks Sam. I started my first startup with Justin Kan right of college. We started a company called Kiko Calendar. It didn't go so well. Well, it went alright. We built it, we sold it on eBay. That's not necessarily the end you want for your start up.

It was a good time. We learned a lot. We learned a lot about programming. We didn't know anything about calendars. Neither one of us were users of calendars. Nor did we, during the period of time we did the thing, go talk to anyone who actually did use a calendar. That was not optimal. We got the build stuff part of the startup down. We did not get to the talk to users part.

The second startup we started, we used a very common trick that lets you get away with not talking to users, which was that we were our own consumer. We had this idea for a television show, Justin.tv, a reality show of Justin Kan's life. We built technology and a website around the reality show we wanted to run. We were the users for that product. One way to cheat and get away with not talking to many other users is if you build something that literally is just for you, you don't need to talk to anyone else because you know what you want and what you need. But that is limiting way to start a startup. Most startups are not built for the person who is using them. When you do that, every now and then you get really lucky and you are a representative of some huge class of people who all want the exact same thing you do. But very often, it just turns into a side project that doesn't go anywhere.

We kept working on Justin.tv for awhile and we actually achieved a good deal of success because it turned out that there were people out there who wanted to do the same thing we did, which was broadcast our lives on the internet. The issue with Justin.tv that kept us from achieving greatness is we hadn't figured out how to build towards anything beyond that initial TV show. We built a great product. If you wanted to run a live 24/7 Reality television show about your life, we had the website for you. We had exactly what you needed but if we wanted to go do more than that. We wanted to open it up to a broader spectrum of people and use cases, but we didn't have the insight to figure that out because we weren't that user.

So we decided to pivot Justin.tv. We decided we need to go in a new direction. We though we built a lot of valuable technology but we hadn't identified the use case that would let it get really big. There were two directions that seemed promising. One was mobile and one was gaming. I lead the gaming initiative inside of the company. What we did with gaming that was very, very different from what we'd ever done before was we actually went and talked to users. Because while I loved watching gaming videos, I was very aware that neither I nor anyone else in the company knew anything about broadcasting video games. I was amped about the content. I thought that there was market there. That was the insight that wasn't common at the time, which was how much fun it was to watch video games.

Quick show of hands, how many people know about watching video games on the internet here? Ok, I am just going to assume if you are listening to this also know about. If you don't know about watching
video games internet you should go read about that, uh, because, uh it's sort of important context for
the stuff I am going to talk about. But, uh, the main point is, uh, I thought that was awesome - but I
didn't know anything about the side of it that was really important which is actually acquiring the
content of the startup broadcasting. So we went out and we ran a - actually a very large number of user
interviews. We talked to, uh, a lot of people and brought that data back and that formed the core of all
the decision making that was for the next three years of product features on Twitch. Uh, was sort of
some of the insights we got from that? And we continued to talk to users and in fact build an entire
part of the company whose job it is basically to talk to our users. Um, which is a whole division that we
just didn't even have at Justin.tv. We had no one at the company whose job it was to talk to our most
important users.

Um, so, uh - so that was Twitch. And I'm going - I want to give you guys a little bit of a, uh - uh, a
little bit of an insight into, uh, with Twitch what that, what that meant going to go talk to users. So,
we determined that the broadcasters were the most important people. Um, and the reason we
determined that was when we went and looked into the market we - I don't - we looked into what
determined why people watched a certain, uh, stream or went to a certain website. They would just
follow the content. Right, you had a, had a piece of content you loved, uh and the broadcaster would
come with you. Um, and that's actually the one really important point about user interviews which is
that who you talk to is as important as what questions you ask and what you pull away from it. Because
if you go and talk to a set of users - if we had gone and talked to viewers only - we would have gotten
a completely different set of feedback than talking to the broadcasters. And talking to the broadcasters
gave us insight into how to build something for them. Um, that turned out to be strategically correct.
Uh, I wish I could tell you the recipe of figuring who the target user is, uh, for your product, and who
your target user should be but there isn't a recipe. Uh, it comes down to think really hard and use your,
use your judgment to figure out who you are really building this for.

Um, so uh, what I want to do is a little bit something interactive now. Which is, uh I'm going a - well I
got a bunch of ideas, uh, from, from you guys actually. So sort of suggested ideas. Um, and I'm going to
pick one of them. And um, I want everyone to sort of sit down and do step one of this process right
now which is think about who would go ask this. Like which people where would you go to find the
people you needed to talk to about this uh, in order to uh, in order to learn about what you should
build. Um, and so the idea we are going to use is - let me see here. Of these ideas - so here the lecture
focused note taking app. The idea is - I don't think that the state of the art for note taking is good
enough yet, um and I want to make a note taking app that uh, improves - you know improves that
experience. Makes taking notes in class better. Um, or taking notes while listening to lecture better.
Um, so maybe it has collaboration features. Uh, maybe it like helps you focus better somehow. It has
multimedia enhancements - I don't know. Right, all sorts of possible features. But that's the, that's the
idea. So take like - take 120 seconds right now and think about not what you would ask or what the
right features for this app is. But who would you talk to? Who should you talk to? Who should the - who, who is going to give
you that feedback that is going to tell you uh, whether this is good or not. I actually mean it right now.
Take your laptop out - Like type - write some stuff down. Think, think about - Like, the - You can - It's,
it's good enough to like think that in your head but actually like if you actually just write it down and
like just come up with the five people you talk to. Of the five types of people you talk to. Um, and who
you think the most important one was. Like actually do it. Because there's nothing like actually running
through a practice of something and trying to do it. It's actually get it into your head that right way to
do it. Um, um I'm gratified to see, hear clicking of keyboards now. Um, if you are following along at
home pause - Actually do it. Think about who would you talk to? Um, because uh, that's uh, that is the
first question, uh for almost any start up that you need to answer is like who is my user and where am I
going to find them?

(Emmett stops the lecture to allow the audience to complete the task he just assigned)

http://startupclass.samaltman.com/ (gekregen van Inne ten Haveinne@darwine.nl 06-42804208) 334 / 412
Emmett Shear: Alright, uh that's like way shorter than you normally use to think about this problem. It's actually really tricky problem and like figuring out where to source the people is pretty hard. But, uh, uh. We're going to move along anyways. In this highly abbreviated version of learning how to build a product and running and run a user interview.

So, um, can I - can I get one volunteer from the audience to come up and uh, tell, tell us what uh - who you would talk to. And we'll talk about it.

(Emmett points to the audience located in the front row.)

Emmett Shear: You guys were all pre-selected. Here you go. (Hands the mike to an audience member) I don't know how to turn this thing on. Here we go. So who do you talk to?

Audience Member 1: Um, I would definitely talk to college students first, obviously because we sit in a lot of lectures. And specifically I want to talk to college students studying different subjects to see if maybe, um - if they are an English major - if that makes a difference versus you studying Math or Computer Science in terms on how you want to take notes in different lectures.

Emmett Shear: Um and uh, so you going to talk to, talk to a bunch of college students. Would you pick any particular subset of college students? Like it sounds like you, we don't, we don't want to talk to all college students? Of like a broad way -

Audience Member 1: I, I want to only talk to college students um like, and break down the divisions by like people who student different areas maybe. And then also maybe it would make sense for people who have like different study techniques. 'Cause some people take a lot of notes. Some people don't take that many notes but still jot stuff down.

Emmett Shear: Right, so I mean - so that's a really good start. Like that is, that is actually - obviously the users you want to go talk to. Especially if you are targeting something at, you know at college students, uh as the consumer. Um, and if you are talking to college students as a consumer, uh the - you are going to get a lot out of students about what their current note taking habits are and you know what they would be excited about. Um, one of the problems with selling things to college students is that college students actually spend very much money. Um, it, it's really hard to get you guys to open your wallets especially, uh if you want them pay for a school related thing. Um, people don't even want to buy text books, right. Uh, I think you probably, probably all use checks or deb or borrow from your friend or whatever. Uh, and so uh, one of the like - one of the things that I think you would be missing if you go after just the students right, is you want to figure out who - who is the most important person, uh to this, to this app? And if you actually had a note taking app - my guess is for colleges, the people most likely to actually buy a note taking app that you guys would use would be college IT. Right? Presumably for the most for the most part you want to sell software to students - like the people who have to be brought into that is usually the school administrator. So I don't know if that would be one approach. I feel like they will - you - presumably go talk to the college students and you find out uh, they don't actually buy any note software now at all. I mean, likely. Uh, mean it's possible they do, uh in which case I'm completely wrong and so this why you have to go and talk to the users. But, uh, you then have to try and maybe try other, other group's right. So I would talk to college I - I would talk to IT administrators as well. Um, I think that's - that's another uh, area that's really promising. You might talk to parents. Right? Who, who, whose spends money on their kids’ education is like willing to pull their wallet out? Like you know the parents of kids? Um, of kids who are freshmen going up to college for the first time. You need this app to make your kid productive so that they don't fail out of
college. Um and - and there is actually a lot of groups that are potential - that aren't necessarily obvious user but who are critical, critical to your apps success, potentially. Um and when you are - at the very beginning of a start up like this when you're like you have this idea that you think is awesome, uh and you want to have that broadest group you possibly can. You don't just want to talk to one type of person and learn that. You want to get familiar with the space. You want to get familiar with the various kind of people could be contributing.

Alright, so, uh - uh - Let's have someone come up and we're, we're going to pretend uh - uh we are going to run this user interview. So we are going to talk to a college student um and try to find out what we should built. You know what we should get, uh, into this note taking app. So, so, so another volunteer please for running an interview.

**Audience Member 2:** Here?

**Emmett Shear:** Yes - Alright so, uh - Hello

**Audience Member 2:** Hi, I'm Stephanie

**Emmett Shear:** Hi Stephanie.

**Audience Member 2:** Nice to meet you.

**Emmett Shear:** Welcome. Thank you for agreeing to do this user interview with us.

**Audience Member 2:** (chuckles and turns back to the audience)

**Emmett Shear:** So, uh, I want to - hear from you about, you know what your note taking habits are. Like how do you take notes today?

**Audience Member 2:** Sure, so um, I take notes in a variety of ways. I like to, um, now because of speed and efficiency and just to come back to it later - it's easy for me to just take notes on my laptop. Um, so a lot of those would be primarily text based. Um, but in certain classes - so for example if I am taking a History class most of it would be in text. But if I am taking it, taking a Physics class for example, there are going to be more complex diagrams for different angles I have to draw. Uh, and so that's a little hard - harder for me to get the...

**Emmett Shear:** What software do you use for this stuff today?

**Audience Member 2:** I just do pen and paper for that.

**Emmett Shear:** You do pen and paper. So you do a combination. You take notes with pen and paper. You take notes with

**Audience Member 2:** Exactly

**Emmett Shear:** the computer sometimes

**Audience Member 2:** Yup
Emmett Shear: Um, and uh, when you take notes with uh - When you take all these notes at the end - do you actually review them? Like do you - Be honest! Do you actually go back and every actually look at this notes?

Audience Member 2: The pen and paper not so much. But yes to the uh, um software based. Because it's easier to access and it's easier for me to share and collaborate and maybe like even merge notes with classmates and friends.

Emmett Shear: Uh, so what do you uh, what did you use to take notes today on your computer?

Audience Member 2: Um, Google Docs? Um and Evernote.

Emmett Shear: Google Docs and Evernote.

Audience Member 2: Um, hm - And

Emmett Shear: And uh, tell me more about - why two things at the same time?

Audience Member 2: Um, so Evernote is easy if I am trying to just collect for myself I think. Um, and yes you can share - but I think Google Docs for me uh, easier to share. And it depends also if you know, a friend has already created folder for example in Google Docs and I just have to add to that folder. If it's a group project for example versus for if it's for my personal use I tend to go more toward Evernote.

Emmett Shear: So it sounds like you have a lot of like note taking collaborations.

Audience Member 2: Yeah, I wish it was integrated. (Chuckles)

Emmett Shear: What uh, (Laughs) what uh, uh - tell me more about that. Like, like do you take a - do you wind up taking most of the notes - most of the value of the notes out of notes that other people take? Or is it mostly your own notes you review at the end of the semester? How does that work?

Audience Member 2: It's mostly mind because I am pretty picky about the way I like things organized. Um, like `design wise or formatting - Um, even color I am really particular with. And like the font that we use and that really effects the way I study. So, um, I tend to like it, to like personalize it. Even after I merge.

Emmett Shear: So you, you pulling notes from other people but then you merge them into, to the main...

Audience Member 2: What works for me?

Emmett Shear: Right. Um, awesome! And if you, uh, if you have Evernote notes and you have uh, Google Docs notes, and you have pen and paper notes

Audience Member 2: Hmm, hm

Emmett Shear: Once the semester is over, do you ever go back to any of that stuff or is it like per quarter? You guys are doing quarters her right?
Audience Member 2: Yes

Emmett Shear: Once the quarter is over, uh, do you ever go back to any of that stuff? Do you ever….

Audience Member 2: Um, for classes not so much. Um, but if its notes that I have taken for like talks like these for example. Or if its interview prep that I am doing, um I tend to go back because its things that I like to kind of keep fresh in my mind and to um, help me prep for future things.

Emmett Shear: That's interesting. Tell me more about that. Like you take notes. Not just in class.

Audience Member 2: Yeah. Um, so I take notes to also just summarize main points. So if it is like inspirational quotes for example from talks that I go to like these. And then and like maybe I am going to some event where I am actually going to meet someone and it helps to actually to think about and to remember and recall um, what was shared at the time that you know I attended the talk or something.

Emmett Shear: Awesome. Uh, well - normally I would actually dig into a lot more detail. Uh, there is a huge amount of like open questions that are still in my mind after hearing that stuff. Um, questions about which people do you collaborate with. Questions about uh, whether or not you - like how long the notes are. And how, how long of note taking stuff. And just sort of digging into what the current behavior is but like an interest of time and not like keeping everyone hear and hearing about the intricacies of one person's note taking habits forever. We're going to move. But thank you very much Stephanie. That was

Audience Member 2: Thank you.

Emmett Shear: I appreciate that. So, uh. So that's like - that kind of stuff. You notice we are not talking about the actual content of the app at all. Like I'm not - I'm not really interested in features. I don't really want to know about, uh what specific feature set in Google Docs or Evernote. I might sort digging in a little more into which features actually get used. Like if she's actively collaborating, you know - how does that work? I heard some interesting things, " Oh yeah we used folder." That's interesting to me. Um, but the main thing you're trying to do when you're running this first set of interviews is not necessary get like - questions about like user flows and like optimizing that. Or questions about uh, like the specifics of um, of any of that stuff. Uh, kind of can be distracting. Because users think they know what they want but like you, you get the uh, the get the horseless carriage effect where you're, you're uh, uh you getting asked for faster horse instead of trying to design the the actual uh, real solution to the problem. If you start asking people that feature. So you want to stay as far away from features as possible because the things they tell you wind almost feeling overwhelmingly real. Uh, when you have a real user asking you for a feature it's almost - it's very hard to say no to them. Uh, because, uh, here's a real person who really has this problem. Their saying build me this feature. But as you start to talk to lots of people and really get a sense for what, what their problems are you figure out if this is actually a promising area or not. And like, based on what I heard there - it's like starting from that user interview, I'm not necessarily positive there is a problem - Or at least there's a, there's a big enough problem that it's worth building a whole new product for. Uh, because I didn't hear a lot like things where, where, where there was a big blocker. Where there is something really wrong with the way, the way it was working. Unless I have some big idea, uh I would take that as a you know maybe a negative sign. Um, but that doesn't necessarily mean you can't, um you can't move forward and keep talking to more people. Because just because you talk to the first person and you don't get anything out of it doesn't mean there is not going to be, uh, a ton more people who actually have a problem. And you, once you've talked to about six/ seven/ eight people you

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
are usually about done. Uh, it's unlikely you're going to discover a bunch of new information there. Um, which is why it is important to talk to different extremes of people. Right? Go, go find people who are different, uh different points. Because you have to six or - six or seven Stanford College students you are gonna get a very different response then if you talk to six or seven uh, high school students or six to seven - or six or seven parents. Um, alright let's go and look at the....

(Looks at notes on phone)

So, um - based on that though, right - uh, I think the - I think it's possible you can come up with, uh a set of ideas. Right? You have this information about how someone takes. You've, you've come up with uh, potentially when you came up with this idea you had, and you had some ideas as to who - When you heard this idea, you had some ideas as to like how you could build something cool. Um, and so if you are going to build just one feature on top of Google Docs, uh, what would that feature be? Right? And that's - for, for, for a new product like this, it might good way to like get started thinking about where to go. Which is, uh - Ok so they are extensively using this thing right now, how can we make that experience just one quantum better? Something that would be really exciting to this person to be one uh, one step ahead. And so, only take two minutes right now and think about what that feature might be. Uh, actually like try to, try to come up with, uh, what, uh what you might do based on what you heard from uh, from Stephanie that could convince her to switch away from her current collaborative multi-person, all working together work flow on Google Docs to your new, your new thing that has all the features as Google Docs plus this one special thing that is like going to make it, uh - it's going to make it bet - uh more useful and convince them to stop using the thing they are already using.

(Emmett stops the lecture to allow the audience time to think)

Awesome, alright. So I am going to invite our third guest if you, you have something now. Uh, I don't, I don't want to put you on the spot if you feel like you're not, and you're not sure. But uh... Yeah

**Audience Member 3:** So what I, what I - is it on?

**Emmett Shear:** Yes

**Audience Member 3:** What I thought about was like the, the reason she uses Evernote is because like, of like sticky note type notes? Like more thoughts and like details. So I feel like Google Docs has like documents? And not like smaller notes. I feel like a feature that would be like super - like a mobile version of drive that doesn't like - isn't that clunky and like doesn't like make you make real documents could like really useful.

**Emmett Shear:** Awesome. So right. That's a good insight. Right? That's exactly - that's one of the things you get out of that user interview and - Now you have this idea. Right. You've gotten this uh, user's feedback. You've got this idea. What if we had a, uh Google Docs that had the collaborative aspects and the group aspects of that but where you uh, you could pull in more little one off notes and it was, it was designed more around note taking. And so the question is now, once you have this idea - Which I think if you, it's, it's uh back and reasonable approach - is this enough? Is this something people would actually switch, uh just to have? And the way to validate that - There's two ways to validate that - One if you are quick at programming you can literally just go build it. Uh, and throw it out into the world and see what happens. Uh and that's uh, that's great and if that when that works, that's uh, an excellent way to approach it. But a lot of the time that one little thing that's just, just a little bit better uh, might take you three months to actually build something that's worthy of actually using. And so, you actually want to go out and validate that idea further before you go ahead and start building it.
Um, so you might take that idea - you might go back, uh go back out and you know, you can sit down with uh, with diagrams - You can, you can draw what the uh, what it looks like. Um, draw the work flow and go bring that in front of people. But uh, the one thing you really don't want to do is ask them - uh, this is just sort of a trap I want to warn you against doing. It's just don't go out and say - if you come up with a feature idea and go out and ask people are you - you know - I got this idea great idea for a feature. Are you excited about it? Because the, the feedback that you get back from users if you tell them about a feature and ask them is this feature good - is often, oh yeah that's great. Like that sounds like such a good idea. Uh, but when you actually take that in front of people, uh, and you actually build it you then find out that well they thought it was such a clever idea no one actually like cares to switch and to get it. And so the one question you can't ask is, is this feature actually good or not. Yes Sam.

**Sam Altman:** What is the minimum that you can do in your experience to actually build in on that question? In asking, you, you know - between asking and actually building the full thing?

**Emmett Shear:** Yeah, so Sam's asking if - what's the, what's the minimum you can actually uh, get away with to validate with - given you can't actually just go and ask them is this good or not. Um, and its, its highly dependent - uh the answer to that is highly dependent on the particular feature. But usually the best thing you can do is uh, is really just hack something together. Right? Is you find - If your, your idea is to build something on top of Google Docs don't figuratively one go rebuild , an awesome Google Documents but for note taking application. Uh, find a way to write a browser extension that, that, that stuffs just that little bit of incremental feature in and, and see if it's actually useful for people. Um, go and like actual - Go, go find a way to cheat is what it comes down to because if you can't actually put it in front of people um, it's really, really hard to uh, to find that out. For bigger things where you are actually trying to, uh get people to spend money - it actually gets a lot easier. So if you are selling, it's great. Sales, is this cure all for this problem. Get people to give you their credit card and I guarantee you they are actually interested in the feature. Uh, it's one of the most validating things that you can do for a product is go out there and actually get them to commit to pay you up front. And the problem is, when you are working on student note taking app that's going to be relatively hard,uh, because you probably - unless your idea is that you're actually going to sell it. It's probably something where you thinking at least if the trial version is free you're not necessarily going to learn uh, that much by trying to charge people money. But, if you go out there and if you, if you can get people to say hey, I am going to give you money - the money test is amazing. It really does uh, clarify whether or not they're really excited about it or not. Because if you're not five dollars excited about it you're probably not very excited about it.

Um, so the last thing I wanted to do is actually work through with you guys uh, what happened at Twitch. Um, so I brought some slides of feedback that I'd like to get put up. It's my only slides for the uh, for the thing. It's, its - what it is its, its representative exerts of Twitch feedback. I had a whole like twenty-six page document full of all the feedback and then I realized that reading that was going to be a little bit tedious and there was no way I could make it through in a lecture.

So, pretend that like this is stuff is all representative of - uh - Like lots of people said this kind of thing out to, to us when we asked them questions and I've already pre- condensed it for you into the real feedback you got.

So, when we were working on Twitch, to go launch it, we uh, we talked to a bunch of existing Justin.tv broadcasters and asked them about their experience broadcasting. What they liked about broadcasting. Why they broadcasted. What they broadcasted. What else was going on in their life? And the usual thing is when you talk to users of your product who are, who detailed users of your product are - they come back to you with actually very detailed things about feature. 'Cause they actually get mired in the
feature and you have to sort of read in between the lines. But, um they ask us for things like “I want a way to clear the ban list in my chatroom.” Like that was actually a very common request. Because there was a very particular issue with how our chatroom is worked. People would ask for the ability to edit titles of highlights after creating them. And, and it's - this stuff was really consistent. As you talked to broadcasters you probably talked to twelve/ fifth, fourteen something like that - broadcasters of the Justin.tv gaming platform. Um, we got all this feedback and you know - What else do we have? We have - your, your competitors have all these cool features like polls and scrolling text. I can, can personalize chat there. And uh, uh - Then we have some positive feedback. Like, oh you guys don't have ads. That's great. Um, and you’re able to ban trolls. So a bunch of stuff about chat. A bunch of stuff around uh, interactivity with, with uh, uh - interactivity with the uh, with their viewers. And that was all really interesting. Right. So this is what the - this is what the Justin.tv broadcasters uh, wanted us to build. And this is what they - what - where they felt pain using, using the product. And uh, so if you thought that what we did was go and address these problems, you would be wrong. Because actually people who are using your service already and are willing to put up with all these issues - kind of means that these are probably not actually the biggest problems. Because if you are willing to ignore the fact that you can edit the ban list and the titles are editable - There is no way to get trolls out of your channel and you’re using the service anyways - Maybe those aren't huge problems. And so that sort of brings up a really important point - is that you have to compare uh, you have to compare groups of people. And compare the level in which they uh, uh argue -

If you go to the next slide um -

Yes. Nice. Uh, we have competitor broadcaster feedback. Which is really interesting. So, this is stuff that we sort of heard a lot from people who are using other broadcast platforms. Um, they wanted to be able to switch multiple people onto their channel at the same time. Uh, they, they complained about us not having a revshare program um, where it talked a lot about how they are trying to make a living and they really wanted to make money uh, pursuing this uh, pursuing this gaming broadcasting thing. They talked about video stability. Our service wasn't good in Europe, Um, specially - but just globally, video stability was this huge, huge issue for them - Um and if you compare and contrast actually - it was really different. Like the things that people who didn't use our service said about what they care about was completely different than what the people who were using the service cared about. Um and we focused on this stuff because this was the stuff where - it was so bad where they weren't even willing to use our service because of it. Um, most of them actually hadn't thought about it because our user base happen to be a very uh, well-educated user base in this, in the area who knew about all their options for, for this. They would, they actually you know, reaching out to them uh, meant that they probably already tried all four services and actually had an opinion. Its great when you can get users who are that, that informed and that, that they don't understand the space that well. Um, and uh, if you go to the - I'm just going to go to the next slide.

Yeah - Here we go. Um, the other big thing we did that I thought was really important was we talked to non-broadcasters. Umm, so we went out there and talked to all the people who weren't using us or competitors. And in many ways those were the most important people. Right? Talking to your competitors - that's a short term win. Right? When someone is using a piece of competing software um, unless you, your piece of competing software is something like Google which is a search engine which everyone uses - ok maybe then, then there are no nonusers to convert. In the case of gaming broadcasting, almost everyone is nonusers. Right. The, the majority of people you are competing with are nonusers. Um, they are people who have never used your services before and they say is actually the most important. What they say is, is the thing that blocks you from expanding from uh, uh - Expanding the size of the market with your features. Right? If all you do is look at your competitors, uh and
Lecture 16: How to Run a User Interview

yourself and all you do is talk to people who use your competitors products - people who use your products - you can never expand - Well not never. You're not learning the things that help you expand the size of the market. You want to talk to people who aren't even trying to use these things yet. Who, who've thought about it maybe but who aren't who aren't in to it. So what did they say um, my computer isn't fast enough? Um, I am focused on training twelve hours day for the next tournament. Um, I like making the perfect video and like editing it. Uh and I upload a couple of things to YouTube. I don't do live streaming. I don't I, I uh I have no desire to go into that space. Um, of uh - and this is actually a particularly - in Korea this is a big problem. Uh, once our strategy gets broadcast in major tournaments we have to start over. We have to like come up with an entirely new strategy. And so the last thing we ever want to do is, broadcast our practice sessions - are you crazy? Uh, that's going to hurt us in the next big tournament.

And so this became, became a big outreach program for us. Trying to figure out how to get people over this. We brought people computers. We uh, we worked really closely with gaming broadcast software companies to help the people who made the broadcasting software - to make that better. Um we started building broadcasting into games and into platforms. Like we built broadcasting into the Xbox. We brought broadcasting into PlayStation 4. Uh, because we had a need to overcome this issue and it was like it was too hard. Uh, broadcasting wasn't uh, wasn't possible. Uh and you sort of combined these for us these were the three big groups we looked at for broadcasting. Uh, and you combined that feedback and what it tells you is not features to build. Right? Because the features they asked for uh, things like polls, um things like a uh, um you know the ability to have like child account - like child accounts on your account. We haven't built most of that stuff. Um, but was was in important were the issues - like the goal they were trying to accomplish there. People wanted money . People wanted stability and quality. Um, people wanted universal access for viewers all around the world, um, to be able to watch them. And so that became our focus actually and we dumped almost all of our resources into things to uh, no one ever mentioned, uh in an interview. Those were the things that actually addressed the problem. And the way that you can tell that it worked is as we would build these things and we would go back to exact same people we actually interviewed and we would say hey - you told us you really cared a lot making money. Well built you this subscription program that will let you make money. And uh, it, it, it's astonishing because most people aren't - had actually never had that experience actually. They've never talked to someone and uh, said it would be really great if your product had feature X and, and then like two months later or a month later your product actually has feature X. Um, or at the very least a feature that addresses a problem that they brought up and so - uh, it was actually the, the people we converted first to our product were the people that we actually talked to about user research. The were the ones who were actually the most impressed. Which is kind of fun. Uh, but it really worked. Because we picked people who were representative. And we picked big broadcasters. Small ones. Medium ones and we, we made sure were addressing their concerns. And that, that was completely different from how we approach the problem on Justin.tv. Because with Justin.tv when we tried to do this we'd - we sat down - We trolled through huge amounts of data. Like we spent tons of time looking through Google analytics. Looking at mix panel. Looking at in house analytics tools. Figuring out how people are trying to use the service. Looking where our traffic came from. Uh, completion rates on flows. We spent all this time doing that and that's good. I mean you can learn things from that. I'm not telling you not to look at your uh, at your data. But, uh, it doesn't tell you where you need to go. It doesn't tell you where, what the problems are you need to address. Um, so we would just sort of invent these ideas in Justin.tv and then nine times out of ten - without taking to someone that idea turns out to be bad. Um, and that's actually one of those disappointing things about doing user interviews and user feedback. Which is why I think so many people don't do it. Which is you're going to get negative news about your favorite pet feature most of the time. Like you're going to have this great idea and your gonna talk to a user and it's going to turn out that nobody actually wants that. Like no one's actually - there are actually completely concerned about a completely different things and they don't care about what you

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
thought was important at all. And uh, that’s a little bit sad and just think about how sad you’d be in four months when you launch that feature and it turns out no one actually wants to use it.

So I think that’s about it for my - the lecture section of what we were talking about. I want to take some questions from the audience.

**Audience Member 4:** What do you see startups get most wrong about interviews - Like most startups don't do it at all, but the ones that do, what are their most common mistakes?

**Emmett Shear:** Um, actually the most common mistakes are showing people your product. Um don’t, don’t show them your product. It’s sort of like telling them about a feature. Um, you want to learn about what’s already in their heads. You want avoid putting things there. The other thing is uh, asking about your pet feature direction. So, if you think you want to add, ad subscriptions to your product - going and asking people would you pay for a subscription? Going and asking them would you use this feature? Um, and I’d say the uh other big mistake people make is talking to who's available whether than talking to who they need to take to. There’s certain users are really easy to get at because they are say - members of your forum already. You know, you have some product forum. You talk to the users on that forum because they are - they’re easy to access to. Um, we had - we spend like weeks digging for identifying information and figuring out who these people were. Uh, so we could contact them so we could talk to them. Um, because a lot of these people weren’t - because it wasn't obviously there were some just user on a site and that site did not support messaging. There was like no obvious way to interact with them. Um so we spent a bunch of time trying to network and find those users and bring them on. Because if you, if you just talk to who’s easy to talk to you’re not really getting uh, uh getting the best data. The fortunate side there is almost everyone is flattered to be asked what they think and uh, so - most of them will actually talk to you and tell you things.

**Audience Member 5:** How hard was it to get buy in from the rest of your company. I mean like you can go and be whatever I’m in charge so you doing what I say but that's probably not the best way of doing it? So how did you get them to?

**Emmett Shear:** That’s a good question. So the question is uh, how hard is it to get buy in from the rest of the company and how do you do it? Um, getting buy in - if you just go to them and say - Uh, I figured, I talked to the users. I figured it out. Uh, we have to build this. Uh, is really hard because people don't trust you. Uh, there's something magic about showing them the interview though. So I really recommend you record interviews. Um like recording interviews is like magic. A - Its stop you from taking notes in the middle. And taking notes is a little bit disruptive. It, it makes it hard for you to feel like you are actually engaged in the conversation. Um and B - you can then play that recording for people. So when - they don't have to be there for the entirety of all the interviews but when you want to make a point about what what, what we should be building and why - you can just play back for the rest of the company that interview and it's like magic the influence it has on peoples thoughts on what's the right thing to build is.

**Audience Member 6:** So did you um - Since you mentioned recording did you uh, try insist on doing Skype interviews whether then over email or what was your impression of....

**Emmett Shear:** Um, yeah - so you definitely want to do Skype, uh or. Sorry so the question was um, do we insist on Skype interviews for recording? Um, you don’t want to do interviews over email if you can avoid it. Uh, because interviews over email are non-interactive. And the most interesting things learn comes from the, "Interesting. Tell me more." Uh, the instantly that you hit - you have this vain in you - they will say something that you didn't expect and the instant that you didn't expect or didn't already
know you should drop into detective mode. And detective mode is - "Huh, that's interesting. Can you tell me more about that?" Um, people don't like silence so they'll keep talking to feel the void. Uh, and the best part about doing over Skype or doing it in person is that you have that interactive uh, feedback and you can actually pull a lot more out of people. Email interviews, they are ok but they are basically useless. If you are person or over Skype they are actually also easy, easy to record. Um, make sure you ask them if it's ok to record it. Uh, it's not polite to record people without their consent but if they are willing to like give you a user interview they'll probably willing for you to record it as well.

**Audience Member 6:** Sorry but what about the international market. Like you mentioned that you have um a lot of users in Korea and I don't know like maybe they don't feel comfortable with English or...

**Emmett Shear:** Yeah. Um, so uh, the question is what about people in the international market where you are trying to do user interviews with people who don't speak your language. That's just really hard and actually to this day Twitch works way better in English speaking Countries then it does in non-English speaking Countries. And I think a big part of that is we are much better than talking to people in English speaking Countries and learning what their needs are and we are not as good at in other Countries. We've tried to address that by hiring people who who speak Korean. Um, having them translate. We've tried to address that by uh, finding representative people in those Countries who speak both English and Korean and reaching out to them. But the problem with that is like you're not actually getting representative sample - no matter how hard you try. The very fact that they are fluent English speaker means they are not representatives of all the people who don't speak fluent English. Um, it's just a hard problem. Uh, it's why companies find it easier to build markets in their home, in their home Country. Uh, much more easily then the abroad. Because it's really hard to talk to users abroad.

**Audience Member 7:** Um, what channels do you use to reach out to them? And do you ever compensate them?

**Emmett Shear:** Um, so the channels we used to reach out to them - What channels do we use to reach out to them? And do we every compensate them? The channels we used to reach out to them uh, were, uh onsite messaging systems. So like if your - most sites websites have some way to contact the user. So if they are a visible user of another website, you use that sites messaging system and say, "Hey. I was watching your stream..." Or whatever this person was doing on the site. "Uh, I loved to ask you some questions about, uh your use, um. Would you mind hopping on a Skype call?" And as for another thing we do is find out whose people were and we send them email. Um, we'd like run into them at events. 'Cause a lot of these people go to the same events and we would like go to the events and like - We wouldn't run the user interview at the event. But you'd get to know them. You would exchange business cards or you know, whatever it is you would actually do now - that isn't business cards. And uh, uh you would get in touch with them. We tend to not to compensate people. Uh, I think if you - If people don't care enough about the problem to like to someone who is trying to solve it, uh, you're probably barking up the wrong tree. We never had any trouble of getting people to talk to us without paying them.

**Audience Member 7:** What about onsite user feedback tools? Do you get no feedback on that?

**Emmett Shear:** So, so it's this whole second set of user feedback that's really important that I should talk about. Um, the question was what about like onsite user feedback tools? Um, and I think the stuff you're talking about where you have like uh, uh a new product and you want to see if it's actually going to work or not. And so you put it in front of people and see how they use it or not. That's really important. That kind of work is super, super important. It can tell you lots of things about where you went wrong building something before you launch it. Uh, which is great. It doesn't tell you what to
build. It helps you iron out the kinks and edges of the thing you did build. But, generally speaking, we uh - that wasn't the user feedback we were getting. I mean that stuff's good - it's good. It's like uh - it's much more similar to though to the data driving approach. Right? You're finding out why people are dropping off of this flow. You're not finding out what problem should I really be solving for them? And what do they care about as a human? And for this kind of like really early stage of the user interview is the kind of user interview which is the kind of interview crucial startups do, uh, that's the - that's where you want to focus. So we didn't bring anyone onsite actually it was almost all over phone or Skype.

Audience Member 8: So for the trick of finding groups of people that can give different kinds of feedback - so as a startup you end up kind of process - is there a group that would focus on first?

Emmett Shear: Yes, for the three different kinds of people - to be focused on one of them, uh, given that we had very limited resources, yes. Uh, we focused on the competing uh, people using competing products. Because we knew that they were already interesting in the behavior that we needed and uh, they were willing to do it at all. And, uh therefore all we had to do was convince them to switch which is a much easier thing to do then to try to create new behavior or non-existed before. Um, and we had to do that because we had to get some quick wins because my gaming project inside of Justin.tv would have been killed if wasn't showing twenty-five percent month overrun growth every single month. So, uh, we did and that meant focusing on short term get the people in right now. And that turned out to be good in generally because uh, it turns out that building something that some people that want generally generalizes uh, and so I want to bring in people who weren't even users of the service as well.

Audience Member 9: This is going back to the beginning - sort of build up for example for the Video Game industry - in the beginning this industry was much decentralized. Like there wasn't a lot of cohesion - like different video game companies consolidating where to turn in selling stuff? But now it's very different. So you said originally you spoke to like broadcasters and uh, you know streamers themselves. How does that change when like, for example like Wyatt has like banned users or professional players from streaming their own stuff. Did you try to gain leverage with that, uh....

Emmett Shear: Yeah, so the question is what about the Game Publishers basically? Right? The Game Publishers is this huge important people in the space. Um, A - the Game Publishers - any big company for that matter isn't going to give you the time of day as a small startup. Um, which is both good and bad. Uh, it means you don't really need me to talk to them because they're uh, they're not interested in you. But it means you actually just can't talk to them. I mean we tried but no one wanted to talk to us. And uh, uh - they did once we started getting some traction and becoming a little bit - slightly bit of a player in the space. I don't really want to talk that bad about them because they - they, they were nice about, enough about it. It's just that you know when you are a tiny little startup. There's lots of tiny little startups and they don't have the time to talk to all of you. Um, as we've gotten bigger actually uh, the point that you know - Game Publishers have become increasingly important uh, conscientiously for us. And if I was to talk about who Twitch does user interviews with now - who will pulled information form now - Uh, it would include Game Publishers. Definitely! Uh, because they've become much more active in the space. It was something that they weren't uh, particular active three or four years ago as much as they are now. And uh, uh that's the really important user interviews in general is that the pool of people you care about is going to shift over time. Uh, the people who get you started - like the crucial people to get your product started for the first six months are not who will be using it three years later. And it's very important you keep doing this stuff. Because one of the things that really easy do is to do a little bit of it in the beginning and a achieve some of a little success and then just sort of stop talking to new people. Um, that's a good way to make the next set of features you build be not as good as the first ones.
**Audience Member 10:** How do you give good user feedback if you're a user?

**Emmett Shear:** Umm, so how do you give good user feedback? Is really a question. So, uh, I think what I, what I want a user to do is I want a user to tell me about uh, what they, like what they are really thinking. Right? And what, what, what their problems really are. And to just sort of ramble. Like I was someone to just like tell me about stuff in their life. Because the, the more you learn about them as a person and sort of them, their - what's going on in context in what they are doing, uh, the easier it is to understand why they want the things they want. And that's really the critical question. So, I'd say like you know what I am looking for in a, in someone when I am doing a user interview is someone who is going to be willing to talk a lot and be willing to give me, give me a full picture. Um, so that's what, what I guess on flip side if you want to be a good - if you want to help people out with good user interview feedback uh, uh, ramble. Like just talk about stuff. Uh, and everything.

Alright great. Well thank you very much!

**Sam Altman:** Thank you very much!
17. How to design hardware products
Lecture 17: How to Design Hardware Products

http://startupclass.samaltman.com/courses/lec17/

Hosain Rahman

Very exciting and thank you Sam for having me. Sam and I have known each other for a long time. We were at different companies and we met in in the early days when he was on his company journey. He asked me to talk today about the hardware journey of building products. What I wanted to do is to give you guys a little bit of an overview of the job: what we do, how we think about the world, and how that informs how we build products, specifically in the process of how we design, how we develop. That all comes together through what we do to change categories.

I always like to start with sort of the broadest thinking. The way we look at the world, um is we think of ourselves at this intersection of really crafted innovation, um, in engineering that's almost invisible to the user in terms of his functionality. And that's at the intersection even beyond design. We have been doing design products for well over decade now. We think of it is going the conversation is shifted even beyond design into beauty. It's that intersection of engineering meets beauty and the whole is to help people with a better life with technology. Um, and largely speaking we do play in this world of what everyone is talking now is "The Internet of Things". We were there much longer before there were such a manicure. Um and the way we that is its smart devices that have computing power and connectivity with sensors in them that are measuring all kinds of things. Their wireless connected and they're all talking to you. And we started on this journey really, really early. Right actually out of engineering school here. Uh, and we were developing core technology. We decided to build consumer products around that. Our first consumer product was the headset. We sort of created um, a headset that became a wearable computer. It was the first traveling headset. Um, that was when we started thinking about wearable computing. We then um, invented the wireless speaker space. Um, around Bluetooth and audio. And I will talk a little about that journey. And the most recently we focused our attention on this sort of this wearable health revolution and using a lot of the sensors that we did in the first generation of headsets and applying them to other parts of the body to understand more about users. So, our view of this world having been here for a long time is that it is a little bit of a mess. Um, in the internet of things. Everything is, is well is smart and connected and everything has an app built to it, that doesn't mean that it is easy for users. Right? Your microwave, your refrigerator, your car, your Xbox - your Xfinity Comcast - Everything has an app. It doesn't talk to to each other. It's really confusing for the user. So we think that there's a desperate need for an organizing principal around all of this. And this is sort of the core of when we start to think about how we build and opportunities to sort of create products. We think about where's the word going? And so if there is going to be such a world that, that everyone's talking about on the internet of things which is happening - You do desperately need these organizing principals so that it's easier for user to understand how to come in and interact with these services. So we think that thinking is the shift from less about the actual things to being about the individual user.

And so ultimately when everyone's talking about wearables and you have things like Google Glass and you have the stuff that we do and Apple Watch - and all this stuff, ultimately what we believe is that when you have things that are on your body 24/7, they become this kind of perfect context engine for everything in the world around you. So my phone is not on me. It's in my jacket or sometimes on the charger. But my Up is on me. And it, it understands everything that is happening when it's tracking my heart rate. It's tracking my respiration. It's tracking all these different things. And when I, when I say context engine - I know I can tell the smart thermostat my nest that I am hot or cold. That device doesn't have that understanding. I can tell it's hot but, that device I'm hot because I'm sick. I went for
a run. It's hot outside. I can tell your car that you're falling asleep. That you're agitated or irritated. So this ultimately where we think the world is moving is that wearables is going to be the center of this revelation around everything being connected and smart. And then we are going to drive what a lot of those interactions are going to be and how they're going to work. Right? Do that's sort of the first principal that we think about when we start to think about - Ok. Where are things going? And what we should, should we build and how think about new categories. Right?

So in order for the vision of what we are talking about to happen, you actually need to be great at almost everything. We need to be great at what we call the full stack. We have to be amazing at building hardware. And these are hardware experiences that people have to keep on all the time. Right? You have to wear them 24/7 'cause if you don't, then everything that I am talking about is sort of a castle in the air. Right? You can't actually create a service that people engage with or get lots of data of it to then go power all these other things if it doesn't start with great hardware. So that's where we start - we try to build, you know these magically experiences in hardware that absolutely power by software. We have developed, um world class software application expertise. Um, we've got to be a good there from the engagement perspective as something like an Instagram or WhatsApp. And then on the data side we got to know what to do with this, this massive amount of information and process it push it and have it work for the user. So we really see ourselves at the intersection for the first time of a company that doing hardware, software and data as three kind of equal stools that have to work together in order to unlock that experience around something that is on you knowing what's happening and then talking to the rest of world around you. And so that's a pretty key piece of what we do. I think it's different then what a lot of other companies doing. And it allows and it requires to play at all levels of of stack. Now this was a complicated thing for us to put together. Because typically you know, people who are great a hardware understand mechanical engineering, electrically engineering. How those things interact. How you build. At scale, how you build tools, etc. They're not typically great at building software and services, right? It's a very different discipline. A very different skill set. So when we first put those pieces together, that created a lot of interesting friction in the company. Our software and application team was so use to moving really, really fast and iterating - where as in the hardware world you've got to take your time because your iteration cycles are much more deliberate. You have tooling that takes sixteen weeks. You can't just tweak stuff and you can't hack it in the same. And so what was interesting to see is we put all these pieces together. The hardware learning to move faster. The software guys thinking more about how they resolve experiences before you actually have to ship it. Versus just throwing something out and A/B testing it. And then the data science sort of informs all of that with sort of more information to make different kinds of decisions.

So, how do we thinking about - how do we go. How do we build products? How do we change categories, First of all - everything for us is a system. Right? We don't think about it discretely just as a piece of hardware or discretely as an application or discretely as a, as platform. We think about it across the whole thing. So this is an example with UP, right. Where we have these tracking sensors on the body out rhythms there it connects to the phone where you have this engaging application service experiences. We use the sensors in the phone. That talks to a lot of stuff we are doing in cloud where we are taking all that information, driving insight on it and then we have a huge platform of thousands of developers - where they're thousands of apps that then plugin and also create more experiences. And so we think about it across the whole spectrum. And I, I'll come back to this system think in, in a second.

So, what is the actual process of creation look like? And this is fun for me because we don't actually talk about this very often. It's quite you know sort of we keep it confidential and private and I know that we are talking
Sam: Not with the entire...

Hosain Rahman: Now we, we're on sort of a live cast. (Inaudible 7:34) So it's fun for me to talk about this for the, for the first time. 'Cause it's not - it's quite deliberate process. Right. And this is a little bit of what it looks like. This is kind of a map where we are much unbridled in our imagination in the exploration phase. We start to validate some of our concepts. Bring those ideas tighter, and tighter and tighter. And then we actually start to build a product. Launch it and then iterate. Right. That's the simplest way to look at it. And I'll take you through each of these steps.

So, in the exploration phase - it is very wild. It's imaginative. We think about the vision of where the world's going. What our strategy is. Which is does the brand stand for. Um, and we think of a lot of, of what you guys do here. You're dreaming. You're imagining it. How do disrupt? What's the future going to look like? Right? It is a little bit science project sometimes. And, and we talk about it in that way. Right. And we do build from inspiration and insight. And that sort of raw creativity. And we want to create and we try to create a form where that's ok. 'Cause a lot times in companies that gets lost. And then we we start to bring into early validation, where I always say, " Like look everyone, when you're doing this stuff, you have to now take those concepts and prove them like you do in PhD Thesis. Right? Where you have your conclusions. You've done, you know, your empirical data collection. And start to say here's where we see it going. And here's what's its going to do. Right. And you outline the story. And then once we sort of sign it off on that phase - we think, ok you know what. There, there this theory's right. We start to go into a concerting phase. Where we start to really think about what is the experience and what's possible. And this is, this is another interesting opportunity for innovation at a more specific level of how this thing will come to life or what it is. Or how will sell that experience. And how will we tell that story. And then it, and then we decide its program. And it goes into a heavy planning phase where we start to look at and say, "Ok. We're doing this. We've got to ship it. There is no turning back."

Right? What are the tradeoffs between all the creativity and all the ideas we want versus what does the physics dictate? What do - you know, battery power or all the different constraints that we have and start making those tradeoffs. And start to look at, you know, how we pull that together. And then we move into um, a development phase. Where it's a hand off between various stages um, and, and really very functional teams in the company. And you pull together and you're solving problems as you go implement this. And you launch it. Uh, and you learn. Um, and you see what users think. And you start to think about where does this stand in that experience continuum that you've been imaging where the worlds going to go. What have we achieved? What haven't? What have we learned from our users? How does that change what we're thinking? And then we start right over again. Right

And so that's the broadest way to think about it. Um, a little bit deeper on the exploration phase, right. So it is VERY much like a building and tinkering process. Right? And, and a lot of it is driven by demo Fridays. Where people kind of have an opportunity to go showcase their work. Because we find that that's a great way to pull it together, pull it in a form where others can consume it and give feedback. Right? And it's a, it's a, it's a really it is a show and tell. Um, obviously hack a thons are a big part of it. There's lots of data that gets driven. And it's lead by our, our Strategic Development Team. Which is traditionally called sort of an R&D Team. And there is participation from product and engineering - both hardware/ software. Um, but they're sort of taking a back seat and they're looking at what these explorations are. Right? And the Executives at the company at this phase are more of a signing board. They're there to poke and pride and tell people, "Hey think about this." Or though... Or did you try that? How does that work. Right?

So and in this phase - in order to move to the next threshold we think about it literally as, "He would I give this guy 50 grand." It's a little bit like an angel investment. Right? Would I give this guy 50 grand to go explore this and see if there's a there there. Is there something to do? And our CTO is the - is sort
of final decision maker. He gets to sort of pick those things internally and say "You know what? I like all the feedback. This is the one I want to go chase down and see what happens." Right?

So then we get into this validation phase. And this where its starts to get really interesting. Still lead by R&D but they're really poking at it and their saying, "How does this work. We have these leadership meetings with the broader cross functional team. I have to show results. I have to go through a scientific process to outline, you know why this works. Why, why is it going to happen?" And then you hear - this is really when we start formulating really an important tool in the company which is what we call WHYS. Defining the WHY are we doing? WHY does this exist? What problem does it solve? I'm going to come back to that in deeper. In, in, in a minute. Right?

Um, so at this point it's still an R&D lead, but this is when a lot of our um, industrial design team (inaudible 12:07) and few project guys. They come in and sort to think about, "Ok how can I pull this concept into something physical? If its hardware. And how's that going to interact with the rest of the pieces of the system?" Our Product Experience Team is still also driving a lot of the core values and what the story boarding is. But it starts to become a lot more real, when we start thinking ok how we will build this. How expensive is this going to be? What's the budget going to be, etc.? Right? Um, and at that point, right - we start to really validate - Can we actually build it. Or do we have wait three years for batteries to be there? Or do we have to wait for this other innovation to happen. Or do we have to wait from a budget perspective. Or whatever it is. Is there a business viability. And then we start to, really start to sketch kind of "briefs. And this where I come in and, and make the final decision. "Ok I think this is. There's really a there there. And we can now take this to the next level and get into - into a play."

And then we go into the concept phase. Right? And this now when the responsibilities shift from the R&D folks to what we call Product Experience Team. And the way we think about product experience at Jawbone is sort of what everyone thinks of as sort of conventional designs. So from industrial design to you know software design. To audio design to - you know anything in that touches that experience. We have writer on that team. Story tellers. We have ID people like Eve who are, um, Genius creators. We have amazing you know app level designers - graphic designers everything. And it's all in one team and we can that Product Experience. And their job is sort of from bits to Adams and back and all the way through to unify us as one organization. Um, and that's when they sort of take a hold of this and they start to really kind of work out what is - what is and this is what we call when you're starting to really drive the WHYS. Thinking about what's possible. There's a lot of innervation and creativity now in the actual implementation of how we're going to build and uh, create a product. And we start to say what the most important things in that product are. What are the most important problems we're going to solve? We call 'me "Hero Experiences." Right? What are we going to do? How - what is the bar that would be acceptable, etc. etc. Right?

Um, and at this point we start to really resolve what we call these WHYS which I will show you again in a minute. Um and why I s that different from anyone else has done? From the competition. From the category. And then what is - where does it go? Right? We don't like to do just one off things. We have to see a broader vision. And this is part of that, that creation experience. Is we look at, at at where do we think the world is moving and how is this thing going to be a stepping stone to that ultimate end vision. That's where the road map starts to get flushed out. Again, I have the ability here to come in answer and be the final decision maker with my team and say, "Yup. We're going to move this to this next phase."

And HERE is also where we look at some of these things. Um, and I want to get into some specific examples. We have fast track programs. Right? So we took - for example, the jam box when we were in this phase and we said, "We're not going to go through another phase. We're going go straight into the
development process. Right? Because we want to get this thing out. We want to test it and mark it. And move really quickly. So we have the ability to sort of, sort of prevent our own process now and say like and let's go really rapid um, and fast track it. Recalibrate kind of the go to market possibility. Right?

So this is when we - after this stage now it shifts from that Product Experience Team to our Product Managers. Who are really defining the business plan? When is it going to launch. When's it going to get into the retail calendar? When does - what is the - is the software release cycle. Really prototyping. Actually starting to feel it. Um, and you're starting to make as I call a lot of those tradeoffs where like - Ok we wanted to build this. We can't do that but here's what we can do. 'We want to be this way. We have to, you know - We want these functional experiences. But we are going to sacrifice battery life. Whatever it is - that's when we start to really kind of pull those decisions and start look at it. And it's really a big juggling act. Uh, frankly at that, that point. Right? And the product guys are driving that - and that's when again, you know, we, we sort of look at and synthesize all of what we put together and we say "Ok. Does it actually cross enough things off our list? Does it meet that, that minimum viability? Right? 'Cause we always start with - as you can tell like this very big wish list of what's possible and what we can do. And then we start to wheel it down and is does this cross enough of the value threshold that we think that it's worth pursuing. Umm, now can we actually move it into the development phase where again Product Management Team continues to lead it. But now you're starting to really get deep and this is where engineering comes in and is really starting to sign off on like we can build it. Here's the time schedule and how we ship. And then again you know, the Product Team is looking at how we should go deeper on this. How can we increase engagement? What are the little innovations? What are the tuning? What are the things that we need to do to sort of make that, Umm, and one of the things that we've been fortunate to have is a lot of wonderful response to products that we've built. And, and we take a lot of care and time and detail. Really in sort of this development and concerting phase around little details that create these magically experiences. So for example when we turn on Jam box, you know we have this pretty cool sound that goes "WOOO". And, that, that took you know, months come up with the right audio tuning. We worked with a lot of different you know choreographers to create that sound but every time someone turns it on I see them smile. And they laugh. The feel of the rubber retyping and there's one manufacturer in the world that was able to make the rubber at the quality of the and drama that we wanted and the colors that we wanted for the first Jam box. Right? So all of those little magical details. How to resolve. Even in software. Right? When we had the first UP and when it plugged in and your sleep graph showed up - Even just the animations of how you know how the bars would show up and the way cards would flow - That was a detail that we thought about. How's this going interact. How's the user going to experience it? How are they going to feel it? Right? So, this - a lot of that stuff happens even at this stage where you sign off on a program. It's going but you're making those kind of decisions all the way through and you're trading them off and you're doing it in the context of, of this bigger picture. So, um and contusing innovation is an opportunity to keep refining and to keep doing all that stuff.

So um, so how do we think about it now, you know, kind of at a broader level? Sort of what is the framework for how we think about these user signature experiences? Well we do start us these WHYS. Right? Which is that articulation of the problem that we are solving. Um and then the themes around how these become actionable concepts. Right? Um, and then there's, there's what we do is we build these cross functional pods that take a person from the Product Experience Team. The person from Hardware Engineering. The person from Software Engineering. A person from the Data Team. And we put them together. They're kind of a person - this is, this is the pod that owns that theme or that track and they continue to sort of build that out against the hero features and the inside features. Um, and how we put that together.
So, I am going to go in now into to more specifics around what we call these WHY’S. ‘Cause this is where I spend, I spend a lot of my time. Where their asking a question. And what it does is it seems as a really interesting framework for us to be able to come back and say "Hey. Did we meet those questions that we asked? Does this thing actually do it?" And it serves also as a really good um, guide post for a lot our creativity and a lot of our innovation so it’s not unbridle. Right? So, it really comes down to a very simple question for us. Which is "What is the user problem that we solve through this experiment?" Whether it's in hardware, software, data, platform - whatever it is that is once we solve it - people can't live without it. Right? They may have an absolutely burring need to go solve this problem and they can't. You know either hey are looking for a solution or its things like - you know once you have it. You never though you needed it but now you can't live without it. Right? Um, and again Jam box is great example of that. We did, you know we talked to some people when were were thinking about making that product and it was interesting because - a little story for you guys... When we launched the Jam box in the fall of 2010, the market for wireless speakers as an overall speaker market was zero percent. Right - zero percent. Last Christmas which was Christmas of 2013 it was 78% of the market. Right? And so in a few years we have transformed this industry that has been around for you know since the 50's and 60's - turn it on its head. And if I gone out and asked a bunch of people who wants a $199.00 speaker for your mobile phone? I guarantee you in that focus group 0% of those people would have said that I want that thing. That I need it and I would be willing to pay for it. But when we did it - it transformed an industry. Right? And so and this where these WHY”s become super important. Just really focusing in on what you're doing. Right? And I'll go through one example in the audio space which is the Jam box example in audio. And then I'll take you through a little bit of how we did it in UP. Particularly UP 24. But its starts with what we call kind of category strategy and this is the sort of experience framework. Right? In our view was look all of your content and media experiences are now in your phone. Right? There are no longer IPad, IPod or your computer. And so we different way to interact with it that needs to be as mobile as portable, as high quality. Right. That was our, our fundamental thinking. Right. And then we said that experience needs to be seamless across time and space. So you could go anywhere through it. 21:19

Sam Altman: Yeah, man questions are great. You got about 15 minutes.

Hosain Rahman: (points at audience member) go ahead.

Audience Member 1: So let's say you have this product, right. You have all these feature that you want to create all these features to satisfy. Um and you're about to enter to the designing process. Right? How do you approach the whole problem? How do you break it down to its going to solve this problem to satisfy this problem? It's going to fix this problem. But then, like each design feature is not mutually exclusive. How do you approach it holistically?

Hosain Rahman: Yes

Audience Member 1 :( 33:17 (inaudible)

Sam Altman: Can you repeat the question by the way.

Hosain Rahman: Yeah and so I think the question was um, when you have a number of different features and functions that you are trying to build - how do you look at it in a system level to understand not within the one cylo from what the tradeoffs are but what the tradeoffs are across the entire system. I think that's the answer to your question. You do exactly that. You don't think about it in one cylo, you have to force a lot of you know, when it's a small team, it's really easy. 'Cause you all are sitting around the table. You're looking at each other. You making those decisions in real time. As
Lecture 17: How to Design Hardware Products

You get bigger in larger company you have to force a lot of communication where everyone is in a room and that person says you know what if you build, if I it, it - if you were to constrain me in this way um, I can't get the quality spec that you need me to make. And this guy is going to say - Well if you do that and you give me that much space then I can't fit all of the out rhythms in at the battery performance that you want. And so when you start to look across the system and you start to see everyone has to have to share what their pains are and they actually understand if I make this trade off its going to affect me over here. And so you have to put them all together in a room and start hashing that. But what's on the board, and on the walls on everywhere - what are we trying to do. Does that trade off still meet it across right across all those, all those different cylos. Because everyone is thinking about the trade off in their bend. They know what they need to accomplish. Right? But again how does that affect the whole, whole thing. We just went through this with UP3 - which is a product we are shipping in a couple of weeks that sort of define the next wave of what happening in the wearable space on the health tracking side. We invented a totally new sensing system. Right? There was RAW science that had been developed that `we productive really fast and even just trade off on what the electro materials were. How it affect so n reliability. Source sing. You know, signal performance. And just - these guys weren't talking their way to get in a room. Do daily calls for three hours where they are going through each of their thing. It's tedious. But we're figuring out and we're knock it down. So it, its - when you're small it's real easy you just draw and look at it. But you have to always have that definition of what you trying to do across the system. That's why a lot of what I was talking about was a much higher level. What problem we're solving. Where does it go? And how all of these pieces are formed.

(Hosain Rahman points at another audience member)

Hosain Rahman: Go ahead

Audience Member 2:(35:35(inaudible) Should we start focusing on one small thing or should it start focusing on the one small thing or should it start moving with the systems itself 35:41 (inaudible)

Hosain Rahman: I think, I think system think system is a way- it's a mindset. Right. It's not actually a system. Right. There are simple systems. There are complex ones. Um. You know. You know a plane is a very complex system. A car is a very complex system. There's other products we make that are most much simpler. Phones is a complex system. An application you should think of as a system. Right? And how all the pieces work together. Storage. You know the front end experience. What you're doing the connectivity. That's all a system Right? And so that's more what mean about system. Yes. For us system is hardware, software and data but I think within anything there's always a system thing. And so it's more just thinking about how the tradeoffs work across all the different pieces that work together. Does that make sense?

Audience member #2: Yeah

Hosain Rahman: (points at another audience member) go ahead

Audience Member #3: another question, um, what's the decision making process between trading not related products and save space for like fitness tracking for different `versions of - or Jam box different versions

Hosain Rahman: Yeah

Audience Member #3: You know - what goes into that?
Hosain Rahman: Um, we do have a grand unified theory. Um, at some point how these experiences come together. And what happens - it touches a little bit how a context engine when you have things on your body that can make everything in the world around you smarter. Like if I know the emotional state of a user, I can tell Spotify what song it should play you on the Jam box. Right? I can tell the TV that you didn't like that commercial and they should fast forward to the next one. Or I can tell you to not watch Game of Thrones on a Sunday night because you don't sleep well. Right? And so you start to see

(Audience laughs)

Hosain Rahman: I'm serious. Right and so these pieces go together. So we do think at that level and so we start to say well what are the building blocks to get there? And how do we establish accreditability and how do we establish a distributing system? How do we establish manufacturing scale? How do these pieces come together?
Lecture 17: Hosain Rahman at Startup School SV 2014


Hosain Rahman

Jessica Livingston: I'm planning on doing not much talking at all. And I'm just going to ask you, because it's a strange thing, I want you to sort of go through this chronologically of the early days. And talk about, you really have had a lot of close calls. I would love to have you talk about them and how you got through them --

Hosain Rahman: What did we call them the other day? Not near death experiences but defibrillator--

Jessica Livingston: Defibrillator experiences; that's great...

Hosain Rahman: Resonate it, you know.

Jessica Livingston: Yeah. I definitely have not heard of so many so I would love to hear a little story. Because it was back in 1999. That was a long time ago.

Hosain Rahman: Yeah, it was 1999. A lot of people don't know this but we started working on the ideas, 1997-1998. For a lack of better way to say it, we tried to build Siri for mobile devices. We were inspired by Palm, Nokia, and others in front of this mobile device revolution. And we thought, "wow wouldn't it be great if you could talk to your phone?" And that would sort of transform and interface that. So we started with this sort of customer problem, and that's where we have done good. When we stray away from that customer problems is where we fold out, and there has been a very of those along the way.

So early on we decided that voice would be the way to interact with this, so how do we make Siri? This voice interaction layer on top of the operating system. And we set out to do that, we realized everything we wanted to do was feature recognition, wasn't going to work. And at the same time we were already started to raise money for the idea, a new way to interact with your device. I say now, it was harder for us to make our first half a million dollars then it was for us to raise our first two hundred million. And it was because in those days, people back in the valley weren't thinking about mobile or mobile voice. Everyone thought yes, there has to be a better way to communicate with their device.

But what's your hook? What's your advantage? What we ended up doing was creating this noise cancellation technology. It turned out to be the biggest breakthrough in audio technology in years. Its this sort of serendipitous thing we stumbled upon. Mostly because we were trying to solve this problem. Then when we did that, we realized this is bigger than making speech recognition, or what people do on their phones which is talk at the time. Now we do lots of other things on our phones but in those days it was a lot about talking, mostly about talking. Then we thought, great everyone is going to want this. Everyone who heard the demo, where we did these A B tests with weed whackers and blenders and all these, everyone was like, "Wow, we want that in our device." And so we set on this journey, we start talking to all the Hansen manufacturers that were big at the time. Nokia, Motorola, the Korean guys, Sony Erickson--Erickson actually, and it was a real wake up call. No one actually wanted to take this demo. Thought, "Yeah it's a neat demo, we think it's cool"--

Jessica Livingston: So they love the demo but---

Hosain Rahman: Yeah they loved the demo, they didn't want to integrate the technology; it was too expensive to add additional microphones. I was with someone from Apple the other day who was telling me that they now have six microphones, I think, in an iPhone and at the time we were just trying to convince people to put an extra one in to support our technology. And it's just like, "Wow, I wonder what we would have been able to do in terms of quality with that." Then again, we were struggling to get our technology into that.

The Hansons were just concerned about the pennies so they didn't want to integrate what we had. So at that time we were struggling with, how do we fund this? How do we take it to the next level? We knew we had a breakthrough and what did we do? So along came DARPA, that was a great thing for us, the Defense Department.

Jessica Livingston: A government agency funded you?

Hosain Rahman: We never thought we would work for the industrial military complex.

Jessica Livingston: What did they want you to do?

Hosain Rahman: They wanted us to make our noise cancel out rhythms in the foundation of their battlefield technology. And we thought. Great. We have this raw science, how do we take that and productize it? Use these government grants to work through all the issues, and take that science into a form that can be delivered into a product? And it was great. Kept the lights on; we were aid engineers. I think we were one of the earliest technology companies in San Francisco. We moved up to San Francisco in 1999 and then there weren't a lot of startups in the city. And I love it now because you walk around, we are in the same block of Zinga, Pinterest, we used to share an office with Air BnB. Its so fun that there's this vibrant community there now, before it wasn't like that.

Jessica Livingston: We need to remind people, we are talking fifteen years ago. It's like the stone ages. Before iPhones, so I want everyone to go back in their mind--
Hosain Rahman: Half these people were probably in grade school.

Jessica Livingston: Totally, let's hope they were all born. So then what happened? You get this money?

Hosain Rahman: Get this start up grant. And the other thing that was really tough was, so again-- it's so fascinating. All the ideas that were kicked around in 1999. I remember Larry and Sergey were RTA's in computer science and Marisa lived in the same freshman dorm room and now she is on the Job board. So it was this crazy time in Stanford, Silicon Valley, and all these interesting people. But you had this huge run up, this infrastructure being laid out for the internet as it stands today. And all these ideas, these delivery services like InstaCart. But none of them survived.

There were online, specific, ecommerce things. For niches, none of these things survived. It was sort of too early. There needed to be a better internet activity, distribution of it, whatever it was, it was just too early for it. The ideas weren't bad, they were just sort of ahead of their time. And I think that's another important lesson that we learned is that you got to stay your course through that. We thought mobile was going to be a big thing. we thought that Palm devices would converge with smart devices. They ended up doing it, ten years after we thought they would do it. But they ended up doing that.

So the early lesson we learned, is if you believe in what you think is right, you think is going to happen in customer experience, the way people are going to interact with technology, then stay that course. Put yourself in a position to do it. That was sort of the first thing with DARPA. We went to a really unconventional source of capital. Right, who had DARPA funding at that time? Very, very, very few people. I just found out recently that Incandme did too. But who remembers Incandme? No one does. They parted a lot of the infrastructure that Yahoo built. So anyways, what happened after the DARPA funding is, the whole technology system kind of cratered. It was impossible to raise money. I remember we used to go up and down Sand Hill Road looking for money. And I remember people asking us, "What's wrong with everyone who went to Stanford from 1993-2000? You guys have lost us billions of dollars. Go get jobs, we don't want to see you here." Its funny because I still run into some of these people.

Jessica Livingston: I bet they like you know.

Hosain Rahman: Its different now, they don't tell me to go get a job in the same way they use to. Again, it's sort of interesting how the tide has changed. It was not a happy, fun place full of ideas. That vibrance wasn't there, there was a really bad feeling around. We had to figure out a way to keep going through that, to believe that mobile is going to be a big thing. And at the time, the valley wasn't the center of the mobile universe. It is now, for sure. And so--
thinking about design. Before that, in the valley, design was a sort of thing that once you got momentum you might put a cool case on something, different color. If you wanted to appeal to women, you put it in pink. And this is what others thought about design.

The only other company that said there was this intersection between design was Apple in 2001-02. Really with the iPod in mass scale. We were fortunate to have such a great tip of the arrow from Apple, going and cutting down the path in front of us. So again, long story short, we put these elements together. We realized how hard it was to take these complex algorithms and to figure out all of the little details that you needed to think about in order to make this product great. Sometimes I sit up and wonder, we make lots of products now, and have been fortunate to ship tens of hundreds of things. But I look at Tusla that makes cars. And I think about all the details we have to think about to get it right and I can't imagine how Elon processes all of that all the time. There are a million things that can go wrong that you have to think about to build a car. And it's not as similar when you have to build these complex products where you have software interfacing with hardware, and now the cloud. And how these things work, there are very little patience people have when things are not working.

Kind of fast forward, we kept launching our first product, which was a headset. And there was kind of a precursor where I knew we might be going wrong on this first product. The first time I was, fortunate enough, to meet Steve Jobs Mid 2004 right before Walter Karsa D Conference. We were going on stage to debut our products and I've been set up to meet him, he is kind of the God of electronics. I went to see him and it was forty-five minutes of absolute, I don't know how to describe it, killed off. He absolutely killed on every single decision, every single edge, every single technology decision.

The crazy part is everything he hit on was right and we knew that deep in our hearts. And we didn’t – because we were trying to meet a time table and because we were trying to sort out of money and ship the thing on time, we sort of made all these trade offs and compromised a lot of things that in our hearts we knew better than. And for me it was a real galvanizing moment, we knew that hey this guy, that’s really good at this is telling me that that was wrong. We shouldn’t have compromised on what we knew was true. It was the first one, I was, shit should have gone with our gut.

Jessica Livingston: So it was harsh but valuable feedback?

Hosain Rahman: Totally.

Jessica Livingston: Okay.

Hosain Rahman: It was unemotional but brutal. Just really, really, really brutal. Just cutting, the only place – I remember talking, this thing was crazy. I don't know if anyone in the audience remembers it. It was a big headset, it connected to a clip on your belt, and it plugged into your phone. And we were arguing about the clip and the product guy at the time that we hired was like, oh no, people will clip this to their belt. And Steve was, the only place anyone would clip that is in your mind. Get it out of your mind, you know. And he was right, we probably sold more of those units sitting here then of that first unit. Because it shipped, it got all this great buzz, we were product of the year in Time magazine. And we sold five of them. Because it had all these problems. We knew there was this really cool technology and that was on the intersection of wait, we didn't stay true to the customer experience and solve the problem we were trying to solve for the user. We compromised all these things to meet schedule criteria, and their real constraints, right?

Jessica Livingston: Didn't follow your instinct?

Hosain Rahman: Did not follow that gut instinct and we knew better. That was probably the hardest thing during my conversation with Steve. And I did send an email to him saying, You were right on everything. I appreciate you taking the time to be honest with me as some people are.

Jessica Livingston: Did he respond?

Hosain Rahman: Yeah, And he was always like that with us. And we launched several other products with Apple, like Jambox and all those things. We had a great relationship with that company and continue to. And we learned a bit more about that experience. So going a bit further on that. We continued to launch this thing that ended up being stillborn. We didn't sell any, ran out of money, and now was like, Oh my God what do we do? And again making a long story short, our investors at the time sort of lost faith and shut the company down. Jawbone was shut down, they laid all the employees off. They literally chained the doors–

Jessica Livingston: They chained the doors to the office?

Hosain Rahman: The chained, locked, I have a letter somewhere saying we terminated all the employees its gone. In those days, too, entrepreneurs and founders, its a lot more friendly now than it use to be from investors. You actually get to control your company. Actually we were kicked off the board.

Jessica Livingston: You were kicked off the board?

Hosain Rahman: We were kicked off the board, we didn't have a say in these corporate decisions. Thank you for your ideas, we will take it from here. And that happened a lot.

Jessica Livingston: So what did you do? You show up and there are chains on the doors?

Hosain Rahman: The thing that I had, and my co-founder had, was we still believed that we found something special. We still believed in the mission. We still believed that we...
could better people's lives and communicate better. And that we could sort of push this forward. So what I ended up doing was sort of wrestle it back from those investors. After we took it back we had sixteen grand in the bank, six grand in debt, and a DARPA contract. We focused on that and said, what are all those mistakes that we made? That we knew better on? And other failure is the best teacher you can have, right? Because you learn so much more. You think about every aspect that went wrong, five hundred and eight degrees again, and again, and again. You ask what went wrong and why did it happen?

Jessica Livingston: Were you independent at this point or are your investors still involved at this point?

Hosain Rahman: They said just go take it. They were already off board.

Jessica Livingston: They lost hope?

Hosain Rahman: Whatever, it was just wasn't even a considered decision--

Jessica Livingston: But it was a blank slate.

Hosain Rahman: We had all this cool technology and had all this promise and people resonated with it. It had to get in the right package, the right form. The right way people could access it, interact with it. We failed on what I feel like design was all about. The attention to the right details, resolving the right details. To make the best type of possible solution for the customer. And we missed that. We missed it in a big way, and it sort of put us out. Again, we focused for the next two years, work for no salaries. DARPA's kind of kept the lights on, as did our friends and families. Then we started thinking about a wearable headset and that is when we thought about programmable computing. And its funny because everyone now talks about wearables, every day. Probably about every hour its mentioned. That is when we thought about all these things on your body and how its wearable and all that type of services. Anyways, thought that it was probably another two days of crazy stories.

Jessica Livingston: Can you talk at all about the shipping story?

Hosain Rahman: Oh you love this story. We went two days with no salaries. We had Vnode and David Whitten at Coastal Adventures help with AT&T to launch our Next Generation bluetooth headset. Everything is lined up, we still can't raise enough Capital to get it off the ground. It was tough but Venod and David saw some problems and were advising us to see what would happen. We were trying to launch, we had product in a dock in New York and the manufacture wouldn't release it to us. We had no money in the bank. we literally had twenty-seven hundred dollars in the bank. And we had to figure out how to get those products released from customers to be delivered to AT&T, who was singular at the time, to get into stores.

Jessica Livingston: In time for Christmas or something?

Hosain Rahman: Christmas! For anyone launching product, December 21st is not a good time to launch product for Christmas. Its kind of too late. And so that was a whole another story. So we still convinced AT&T to take it in and one of our Angels, Chris Birch and Austin Herst, gave us a line of credit to unlock it. I had to go to JFK to find the customers to give them what the harmonized code was to unlock this type of product, and get them to release this stuff to get it to stores. And Mowkes wrote about it in their journal and it sold out in a couple hours. I think we went from zero to seventy million in revenue in our first year. So thats the thing that I learned that I hope they focus on. When you go focus on profusely what those things are that you are trying to solve and those problems, you can transform your business like that. The other stuff, you got to figure out. Its not easy. But thats the core thing. And if you dont do it, everything goes really bad. And if you do, you can build from there just transfer over night like that.

Jessica Livingston: Literally it was like this, over night.

Hosain Rahman: Overnight.

Jessica Livingston: Then what happened?

Hosain Rahman: Then we had to go build a business. we had to scale, build a team. We were fortunate in that as we built out a foundation and built a team. Make the product, it was a success, then we had to scale from there. No infrastructure. We went from zero to seventy million with a consumer hardware product. In that moment in the beginning of January, we had six people. So we scaled that, with revenue it grew to twenty two. Thats when Node and MarketBend, Ben Greson, Sequoia all came in. We started to build the company. Then we had this screaming 2008. Business was growing, growing, growing then suddenly you had the big meltdown of 2008. When Lemon Brother exploded and we found ourselves with millions of products on back order. And all of a sudden those orders got canceled.

We found ourselves with fifty to seventeen million in China. We had to learn and work that down. We spent a lot of time working that down to zero, we did a big deal with Costco to try and move it. So we had all these bumps and bruises and again we experienced in again in 2011. I had announced this new up-product on stage. All these 4A and sensors, heads, then translating it to health. It was this massive, literally debut this thing on stage with Ted. We had a billion of media presses around around it. It was--I still think it was the fastest selling third party product with Apple that fall. They went out there were issues about them breaking at scale. We couldn't even get units back to figure out what was wrong fast enough.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
And here is this incredibly high product, we were thinking about levitating in that product even before we had learned to crawl or walk. And this all type, you don't even know what to do for the consumer- we were fortunate because we have been through some hard times. So there is a DNA in the corporation that we can take a step back and say lets try to figure out what is happening. First principle is to try and take care of the customer. Make sure users are happy, the first question I ask is, Are we doing any physical harm to people? Is there anything wrong? Is it hurting people?

Jessica Livingston: Is it catching fire?

Hosain Rahman: Right, is it catching fire. Is it burning peoples' wrists off? We had one of our competitors--not good. So I think you have to go from there and we are fortunate that we have been through hard times. It was never this straight shot of success. And I often tell people, the things you encounter when you are five people, tens of thousand of dollars in the bank. Compared to billions in market value and big channels--they are less exponential or nerve racking to different scales. So if you learn to persevere from those things and you focus on the things that matter, you will power thought those things for the whole journey. And then you forget those lessons of when you were on the ground with your teeth in the taste of blood in your mouth and how you go through it. And what we did with UP is we found out we had an issue, and I sat down and wrote a letter to people. We made a mistake, we screwed this up. We will give you the money back and you keep the product. We are going to go fix this stuff and we will be back. And when we came back, consumers embraced up and I was surprised with that letter I was getting death threats on Twitter to people saying, wow that is the example of how you treat customers. We weren't trying to think about just what's the right thing to do.

Jessica Livingston: I feel like I would tell people to never start a competitor on any of your products. You are the toughest founder who has been through so much. They would never survive. So we have a couple more minutes and I know there is so much I wanted to get through.

Hosain Rahman: Hopefully that is good.

Jessica Livingston: It was good, we talked about the near death ones that I liked. Can you talk about how hard it is to mix software and hardware data and bring them all together?

Hosain Rahman: Its cool because I feel like everything we have done over the past fourteen years has been a preparation for it. I feel like we have been going to school, where we have learned how to build these great, high design products. And e have been fortunate to be recognized for it. One of the things that was exciting from the first launch is we built our own application. And in 2011 we didn't have an application team, we had to build that from scratch. So one of the things our hardware issue was how crap our app was. It was really bad. All my friends said, why are you building software? I was like, we always have made software just the algorithm and this stuff--it was a really tough learning experience getting people from the application world to work with people in the hardware department instead of working in silos and the some magical point for it to all come together.

Because that actually doesn't work. So I used a lot of 2012 to rip it down and say--what we learned in hardware is you have to make stuff that people will pay you more than what it costs to make. Which is a really hard discipline. you end up being focused on, is this good enough that someone would take a dollar out and pay you for it? In app development and lots of web development people don't have that discipline. Everything is free and moves quickly. So people just figure out stuff. But there isn't that discipline of, is it good?Are we solving that problem? Will people pay us for it? So we took that from our hardware team and tried to get our software team to think about their experiences in that aspect to trying to resolve it.

Yeah to say we are making judgements on where it is going and how it comes together. And then we had the speed that the software team has to reiterate to the hardware team. Trying to get the best of what these things are. And then I organized it again what the customer problem was. We are solving this customer problem. We are going to use these things as tools. we are going to solve it in the software app, or the hardware, or the cloud data. That was another new thing for us is build a data team. I think we have a world class one, we publish a lot of those results. I measured in world class against Google and those guys because they are trying to take out of our team.

Which I think is a sign that you arrived when the big guys try to steal your people. But putting all those elements together and putting them together because they speak different languages is really difficult and I think that is the new tip of the arrow. I think that comes back to that advantage that we have other people and be successful. They are no longer these dumb boxes. There are boxes where the applications experience and software is just as important as what you physically touch. And there is a melding of those lines. And I tell people when we think of design and mobile applications it's something between three dimensions and two dimensions. Its not just a little visual thing. You've got to interact with it and that is not quite three dimensions but something in between. Putting these pieces together. Then again, it's about solving and moving these pieces around to actually solve that.

Jessica Livingston: Well sadly, we are over. Thank you and congrats on all your hard earned success. I'm obsessed with my Jambox Mini and thank you for coming out today.
18. Mechanics: legal, finance, hr...
Lecture 18: Mechanics--Legal, Finance, HR, etc.

http://startupclass.samaltman.com/courses/lec18/

Kirsty Nathoo

Sam Altman: They're going to talk about, uh Finance and Legal Mechanics for Start Ups. This is certainly not the most exciting of the classes. Sorry. But if you get this right, this is probably the class that incites the most gain. Thank you very much for coming.

Carolynn: Like Sam said, this lecture is about the Mechanics of the Startup. Kirsty and I are going to be talking about the basic legal and accounting issues that your startup may face in the very beginning. I was watching Paul Graham's video and at one point he says, "Founders don't need to know the mechanics of starting a startup." And I thought, "Oh no! That's exactly what Sam titled this lecture."

What PG actually says is that Founders don't need to know the mechanics in detail. It's very dangerous for Founders to get bogged down in the details. That's exactly right. Kirsty and I can't give you the details in forty-five minutes anyway. Our goal here today is to make sure that you do know better than to form your startup as a Florida, LLC.

Kirsty: As Sam mentioned, we were also worrying that this was going to be pretty boring for you to listen to an Accountant and a Lawyer talking. You've had some really amazing founders talking about really interesting things. But like Sam said, if you know the basics, you can get yourself set up in the right way, avoid pain, stop worrying about it, and then concentrate on what you actually want to do, which is make your company a success.

We refer to this term "startup" all the time. In the back of your head, you probably know a "startup" has to be a separate legal entity. We're going to talk a bit more about how you actually set that up and what that means.

You also probably know that a startup will have assets, IP, inventions, other things, and that the company needs to protect those. So we'll talk a bit more about that and about raising money, hiring employees, and entering into contracts.

There are a few other things that you need to talk about when setting up your company which ferret out a few issues amongst founders. Who's going to be in charge? How much equity is everybody going to own?

Carolynn: This is us. Kirsty has the calculator. I have the geriatric glasses, which is actually pretty fair.

The first thing we're going to talk is formation. Your start up is going to be a separate legal entity. You probably already know this but the primary purpose for forming a separate legal entity is to protect yourselves from personal liability. If your company ever gets sued, it's not your money in your bank account that the person can take. It's the corporation.

Then the question is: where do you form one? Theoretically you have fifty choices, but the easiest place is Delaware. I'm sure you're all familiar with that as well. Delaware is in the business of forming corporations. The law there is very clear and very settled. It's the standard. The other thing is that investors are very comfortable with Delaware. They already invest in companies that are Delaware corporations. Most of their investments are probably Delaware corporations. So if you are also a
Delaware corporation then everything becomes much simpler. There's less diligence for the investor to do. You don't have to have a conversation about whether or not to reincorporate your Washington into Delaware.

We had a company at YC about two years ago that was originally formed as an LLC in, I'll say Connecticut. The founders had lawyer friends there who said that this was right way to do it. When they came to YC we said, you need to convert to Delaware. The Lawyers in Connecticut did the conversion paperwork and unfortunately they didn't do it right. They made a very simple mistake, but it was a very crucial mistake. The company was recently raising money, a lot of money, and this mistake was uncovered. The company thought it was a Delaware corporation for a couple of years but in fact it was still a Connecticut LLC. I'll just say this: four different law firms were needed to figure that one out. Two in Delaware and one in Connecticut. One here in Silicon Valley. The bill right now is at five hundred thousand dollars for a conversion mistake.

What's the take away here? Pretty simple. Keep it really simple and familiar for yourself. The reason we incorporate all companies the same way at Y Combinator is because it's easy. Don't get fancy. Save yourself time and money.

Kirsty: Once you decide that you're going to be a Delaware corporation, how do you actually set that up? It requires a few different steps. The first one is really easy. You literally just fax two pieces of paper into Delaware saying we're going to set up a corporation. All that does though is create a shell of a company. It doesn't actually do anything within the company. After that, you then need to complete a set of documents that approve the by-laws of the company. It creates a board of directors. It creates officers of the company. Delaware requires that someone has the title of CEO, President, and Secretary. At this point, you also need to complete documents that assign any inventions or any code that you as an individual create so that the company actually owns that. Remember, at this point it's a really good thing to think about, "Am I doing this as an individual, or am I doing this on behalf of the company, which is a separate entity." You have to maintain that split in your mind.

There are services that can help you get incorporated. You can use a law firm, but there are also other online services that help. The one that we often use with YC companies is called Clerky. They are set up so that all that standard basic documents are used and they get you set up in a very vanilla way so that you can move on and keep focusing on what you need to do.

A note on paperwork. You're creating documents. These are really important documents that are going to be setting what the company does and what the company is. It's really important that you actually keep these signed documents in a safe place. It sounds so basic but we get so many founders saying "Oh I don't know, these some documents." And they have no idea what they are or where there are. So really, really make sure that you keep them in a safe place. Let's be honest, filing documents is not the glamorous part of running a startup. The times where this is crucial are going to be high stress times in the startup's life. It's likely when the company is raising a big Series A round or if the company is being acquired. The company will have to go through due diligence and there will be lawyers asking for all this stuff. If you don't have it and you don't know where it is, you're making a stressful situation even more stressful.

The key thing here is keep those documents in a safe place. Keep them organized. It will make your life so much easier.
Carolynn: Now we're going to talk about equity. We're going to touch on a couple different things in this section. The first thing that we're going to talk about is Equity Allocation. If your company stock is high, how to divide the pie. You have to talk about this with your Co-Founders. Why is this important? If you're Solo-Founder this really isn't important. If you are a team of two or more, then this issue is absolutely critical.

The first thing that you need to know is that execution has greater value than the idea. What do I mean by that? A lot of Founder Teams give way too much credit and therefore a lot of the company's equity to the person who came up with the idea for the company. Ideas are obviously very important but they have zero value. Who's ever heard of a billion dollar payment for just an idea? Value is created when the whole Founder team works together to execute on an idea. You need to resist the urge to give a disproportionate amount of stock to the Founder who is credited with coming up with the idea for the company.

The next thing you want to think about is if the stock should be allocated equally among the Founders. From our perspective the simple answer is probably yes. Stock allocation doesn't have to be exactly equal, but if it's very disproportionate, that's a huge red flag for us. We wonder what conversation is not happening among the Founder team when the ownership isn't equal. For example, is one Founder secretly thinking that this whole startup thing is temporary? Has one Founder overinflated the work that he or she has already done on the company? Or overinflated his or her education or prior experience? Do the Founders really trust each other? Have they been honest with each other about their exceptions for the startup and for the future? When ownership is disproportionate, we worry that the Founders are not in sync with one another.

Thirdly, it's really important to look forward in the startup. Said another way, all the Founders have to be in it one hundred percent. Are they all in it for the long haul? If the expectation at your startup is that each Founder is in it one hundred percent, for the long haul, then everything that happened before the formation of the company shouldn't matter. It doesn't matter who thought of the idea, who did the coding, who built the prototype, or which one has an MBA. It will feel better to the whole team if the allocation is equal because the whole team is necessary for execution. The take away on this point: in the top YC companies, which we call those with the highest valuations, there are zero instances where the Founders have a significantly disproportionate equity split.

Kirsty: You've had the conversation about to split the equity but then what? We talk to many Founders who are actually surprised that they have do something in order to own this stock. They think that talking about it is actually enough. This is another situation where you have to think about you as an individual versus you as a representative of the company. And if you equate this to a large company, if you worked Google and you were told that as part of your compensation package you would be receiving shares, you would expect to sign something to get those shares. If you didn't, you would be thinking, "What's going on here?" It's the same thing with a small company.

In this case the documents that you're signing is a stock purchase engagement. You as an individual buy the shares from the company. In any situation, if you're buying something there's a two way transaction. In this case you're getting shares in return for either a cash payment or for contributing IP or inventions or code to the company so that the company actually owns everything that you've done in the past. We also refer to that stock as being restricted because it vests over time. We're going to cover that next, in more detail. As a result of the stock being restricted and vesting, there's one very crucial piece of paper that we talk about until we're blue in the face to everybody because there's actually no way to go back and fix this. This is one of the things that has blown up deals in the past. We've seen companies who haven't filed what's called an 83B Election, and deals have blown up. I'm not going to
go into detail about the 83B Election, but it affects your individual taxes and it affects the company's taxes. It can have a big impact. The main things here are sign the paper work, sign the Stock Purchase Agreements, sign the 83B Election, and make sure that you actually have proof that you sent that in. If you don't have the proof it just goes into a black hole at the IRS. Investors and acquirers will walk away from a deal if you can't prove that.

Carolynn: The next thing we are going talk about is vesting. I imagine that many of you are familiar with vesting, but just in case, vesting means that you get full ownership of your stock over a specific period of time. We're talking about the stock that Kirsty just said. You bought your stock of your company and you own it and you get to vote, but if you leave before this vesting period is over, then the company can get those unvested shares back. When you hear restricted stock, it means that the stock is subject to vesting. The IRS speak for this is, "Shares that are subject to forfeiture."

What should a typical vesting period be? In Silicon Valley the so called standard vesting period is four years with a one year cliff. This means that after one year, the Founder vests in or fully owns twenty-five percent of the shares. Then the remaining shares vest monthly over the next three years. Here's an example. Founder buys stock on Christmas day, let's say, and then quits the company on the following Thanksgiving. So before the year has passed. In that case the Founder leaves with zero shares, because the cliff period hasn't been met. If the Founder quits the day after the next Christmas, so a year and day later, he or she is vesting in twenty-five percent of the shares. In that case the one year cliff has been met.

What happens to the shares when a Founder stops working at the company? The company can repurchase those shares. In the example I just gave where the Founder quit a year and a day after purchasing the shares, seventy-five percent of the shares are still invested and the company will repurchase that full seventy-five percent of the shares from the Founder. How? They just write the Founder a check. That's how the Founder brought it. It's the same price per share that the Founder paid, so it's simply giving the Founder his or her money back.

So why have vesting? Why would Founders do this to themselves? They're doing this to their own shares. The number one reason why vesting is important has to do with Founders leaving the company. If you didn't have vesting and a Founder leaves, a huge chunk of the equity ownership leaves with or her. Obviously that is not fair to the Founders left behind. We're actually going to talk about this a little bit more when we get to the the "Founder Employment" slide. I will go into that in more detail.

The other reason to have vesting is the concept of skin in the game, the idea that Founders need to be incentivized to keep working on their startup. If the Founder can walk away with his or her full ownership at any point and time, then why would you stay and grind away? Startups are hard.

Do solo founders need vesting? They do and the reason is because the skin of the game concept applies to solo founders as well. Investors want to see all founders, even solo founders, incentivized to stay with at the company for a long time. The other reason that solo founders should put vesting on their shares is to set an example for employees. You can imagine it would be inappropriate for a Founder to tell an employee that he or she has to have four year vesting on his or her shares but the founder doesn't think he or she needs any on their own shares. It's a culture point. A founder who has vesting on his or her shares then sets the tone for the company by saying, "We're all in it for the long hall. We are all vesting on our shares. We're doing this together."
Vesting aligns incentives among the founders if they all have to stick it out and grow the company before they get any of that company. Investors don't want put money in a company where the founders can quit whenever they feel like it and still have a big equity ownership stake in that company.

**Kirsty:** Moving on. We've now got a beautifully formed corporation in Delaware. Everyone has their stock. It's all in the plain vanilla standard paperwork. Then what? The next stage of a company's life is to raise money. We know that you already heard a lot from a lot investors and from founders in this set of classes. They've been talking about the tactics of how to raise money but what about the paperwork? What about when somebody actually agrees to invest?

So, first of all in terms of logistics in very simple terms there are two ways to raise money. So, either the price is set for what the money that comes in or the price isn't set. And by price we mean the valuation of the company. It's, it's the same things. So rounds can actually be called anything. People can name, name them whatever they want. But generally when you hear the team seed round, it would mean that the price has not been set. And anything that's a Series A or Series B would be something where the price has been set.

So not setting the price is the most straightforward, fast route to getting money. And usually the way that this is done is through convertible notes or safes. And again this is a two way transaction. So it's a piece of paper that says for example, that an investor is paying one hundred thousand dollars now. And in return has the right to receive stock at a future date when the price is set by investors in a priced round. So it's important to note that at the time that paperwork is set, that investor is not a shareholder. And therefore doesn't have any voting rights on the company. They will have some other rights um, which Carolynn is going to talk about separately.

Of course investors want something in return for putting in money at the earliest, i.e. riskiest stage of the company's life. And this is where the concept of evaluation cap comes in. Which I'm sure many of you heard mentioned before. So usually the documents, um, for unpriced round set a cap for the conversion into shares. And that's not the current valuation of the company. Um, it's actually an upper bound on the valuation used in future to use - uh, used in future to calculate how many shares that investors going to get. So for an example, an investors that invests a one hundred thousand dollars on a safe with a five million dollar cap, um then a year late the company raises a priced round with a valuation of let's say twenty million dollars, then the early investor would have much, much lower price per share. About quarter. Uh, and therefore their hundred thousand dollars would buy them approximately four times more shares than an investor that was coming and putting in a hundred thousand in that series A priced round. So that's where they get there, their reward for being in early.

So again this is another situation where you need to make sure you have the signed documents. Um, and you, you know where they are because different investors may have different. And so you need to know what those things are. And again services like Clerky ca, can help with that. They have very standard documents that most of our YC companies use to raise money on.

A couple of other things to think about when you are raising money, um. Hopefully you got a really hot company that, that's doing great and it's really easy to raise money. But you should be aware that all these people throwing money at you does have some down sides. Um, so the first thing is to understand your future dilution. So, if you raise, let's say two million dollars on safes with a valuation cap of six million dollars, then when those safes convert into equity, those early investors are going own about twenty-five percent of the company. And that's going to be an addition to the investors that are coming in at that priced round who may want to own twenty percent of the company. So you're already at that point given away forty-five percent of the company. So is this really what you want? And you know the
answer might be yes. Um, remember that some money on a low valuation cap is infinitely better than no money at all. And if those term that you can get then, then take that money. Um, but it's just something to be aware of and to follow through the whole process so that you can see where this is going to lead you down the road.

The other thing to bare on mind is that invest, investors should be sophisticated. Um, and by that we mean that they, they have enough money to be able to invest. Um, and they, they understand that investing in startups in a risky business. You know we, we see so many companies coming into us say "Oh yeah. My, my uncle out money in or my neighbor put money in." And they've put in five or ten thousand dollars each. And often those are the investors that cause the most problems going forward because they don't understand how this is a long term gain. And so you know they get to the point where sitting thinking, "Hmm. I could actually do with that money back because I need a new kitchen." Or "You know this start up investing is not actually as exciting as all the TV shows and movies made it out to be." So those cause problems to the company. You know they're, they're asking for their money back. So, just be aware that you should really be money from people who are sophisticated and know what they're doing. And the term that you'll here that, that refers to this people are that they are accredited investors.

So really the main points here is - keep it simple. Raise your money using standard documents. Um, keep, make sure that you have people who understand what they are getting into and understand what you're getting into in terms of future dilution.

Carolyann: Ok. So you're raising money. You understand what you're selling. You figured out the price. You got down the logistics that Kirsty just described. Um, but what you may find is that you don't understand some of the terms and terminology that your investors are using. And this is ok. But, you have a burden to go and figure that stuff out. Um, don't assume that just because you have agreed on the valuation of the price. Um, that all the other stuff doesn't matter because it does matter and you need to know how these terms are going to impact your company in the long run. At YCombator, Kirsty and I hear Founders say all the time "I didn't know what that was." "I didn't know what I was signing." You know, "I didn't know I agreed to that!" So the burden is on you to figure this out, figure this stuff out. And we're going to go over four common investor requests.

So the first one is board seat. Some investors will ask for seat on your company's board of directors. And the investor usually wants to be a director either because he or she wants to keep tabs on their money or because he or she really thinks they can help you run your business. And you have to be really careful about adding an investor to your board. In most cases you want to say no. Otherwise make sure it's really a person who's really going to add value. Having money is very valuable but someone who helps with strategy and direct is priceless. So choose wisely.

The other things is advisers. They are so many people who want to give advice to start ups. And so few people who actually give, give good advice. Once an investor has given your company money, that person should be a defector adviser. But without any official title and more importantly without the company having to give anything extra in return for the advice.

So here's an example - um, at YCombinator, we've noticed that whenever a startup manages to garner a celebrity investor, the celebrity almost always ask to be an adviser. We have a company that provides on the demand body guard services. An NBA basketball player invested. Asked to be an adviser and then asked to be given shares of common stock in exchange for adviser services. And the services that this person had in mind - this investor had in mind was to introduce this company around to all the other professional basketball players he might want to use an on demand body guard. But this celebrity just
made a big investment. Um, shouldn't he want to help the company succeed anyway? Why does he need something extra? All investors who can help - should do so. Asking for additional shares is just an investor looking for a freebie.

Ok, next we're going to talk about pro rater rights. What are pro rater rights? Some of you may have heard of this before but very simply it's the right to maintain your percentage ownership in a company by buying more shares in the company in the future. Pro rater rights are a way to avoid dilution and dilution in this context means owning less and less of the company each time the company sells more stock to other investors.

So - this is a really basic example, but say an early investor buys shares of preferred stock and ends up owning three percent of the company once the financing has closed. And the company raises another round of financing. And the company will go to this investor who negotiated and go pro rater rights and say, “Hey. We're raising more money. And so you're welcome to buy this many shares in the new round to keep your ownership at approximately three percent. That is pro rater rights at the very most basic.

So pro rater rights are very common request for, from investors and they are not necessarily a bad thing. But you absolutely as a founder need to how pro rater rights work. Especially because - Kirsty touched on this a little bit - the corollary to an investor having pro rater rights to avoid dilution is that founders typically suffer greater dilution.

The final thing is information rights. Investors almost always want contractual information rights about, to get certain information about your company. Um, giving periodic information and status updates is not a bad thing. In fact at YC we encourage companies to give monthly updates to their investors because it's a great opportunity to ask for help from your investors like introductions or help with hiring. That kind of thing. But you have to be really careful about over reach. Any investor whose saying they want like a monthly budget or weekly update. That's not ok.

So the take away here is that just because the type of financing and the valuation has been negotiated doesn't mean that everything else is unimportant. You need to know everything about your financing.

**Kirsty:** Ok, so then moving on to after you got that money. Um, you know you've raised some money. The company bank account is probably sharing more zeros in it then you've ever seen in your life. So, so then what? Um, this is where you actually start incurring business expenses. And business expenses are the cost of carrying out your business. So things like paying employees, um, paying rent for an office - hosting costs. Acquiring customers. That kind of thing. And business expenses are important because they get deducted on the company's tax return to offset any revenues that are made to lower their taxes that the company pays. And on the flip side if it's a non-business expense that the company incurs and that is not deductible on the tax return so that can increase the profits of the company that have to pay tax on them.

So, again this is, this is a separation issue. Um, the company will have its own bank account and that's where the company's expenses should be paid out of. Um, again think about this from a, from a large company, if you were working at Google, you would not use a Google credit card to buy a tooth brush and tooth paste.

So the other thing to remember is that you know that they investors gave you this money. They trusted you with all these huge amounts of, of, of money. And they want you to use that money to make the company a success. It's not your money for you to spend how you please. And believe me we've had some horror stories of founders who've take that approach. Um, we had, we had one founder that we
knew of who took investor money and went off to Vegas. And boy by his Facebook photos did he have a good time. Uh, needless to say he's no longer with the company. Uh, but really this is, this is stealing from the investors.

Um, think about it - Um the concept of business expenses can get a little bit blurry. Especially in the early days when you're working, working in your apartment and you're working twenty four hours a day. But the way to think about it is - if an investor asked me what I'd spent their money on and I had to give a line by line break down of that - would I be embarrassed about telling any of - telling them what any of those lines were. And if you were, it's probably not a business expense.

So, the other thing to bear in mind is that you know, you're busy running your company at ninety miles an hour just constant, constant - so you don't have to necessarily think about the book keeping and accounting at that point. But it's really crucial that you do keep the receipts so when you do engage a book keeper or a CPA to prepare your tax returns they unpick all of this and they can figure out what are business expenses and what aren't business expenses but their going need your help as a founder because they aren't going to know what all these things. So there is some involvement from you and the way make the involvement the least amount possible is to keep those documents in a safe place. So you can refer back to them.

So, if nothing else that you remember, do not go to Vegas on investors' money. And spend that money wisely.

Carolynn: In this section we're going to hit a couple topics in this section. The first one is "Founder Employment." Why are we talking about founder employment? As we said already, the company is separate legal entity. It exists completely separately from you as founders. As prestigious as we think the title founder is, you're really just a company employee and founders have to be paid. Working for free is against the law and founders should not let their company take on this liability. You wouldn't work for free anywhere else, so why is your startup an exception? Companies have to pay payroll taxes. We had a YC company that completely blocked their payroll taxes for three years. It was huge expensive disaster and in extreme cases, people can actually go to jail for that. Fortunately not in this case, but it's bad. The moral of this story is set up a payroll service. This is something that is worth spending your money on. Don't go overboard on lavish salaries. Minimum wage. This is still a startup and you have to run lean.

Now I am going to mention founder break ups. First, what is a founder break up? In this context, I'm talking about one founder on the team being asked to leave the company. Founders are employees, so that means your co-founders are firing you. Why are we talking about break ups in founder compensation? At YC we have seen a ton of founder break ups and we know that the break ups get extra ugly when the founders haven't paid themselves. Why? Unpaid wages become leverage for the fired founder to get something that he or she wants from the company. Typically that is vesting acceleration. The fired founder says, "Hey. My lawyer says you broke the law by not paying me. If you pay me and you give me some shares that I am actually not really entitled to, I'll sign a release and make all this ugliness go away." If you're the remaining co-founders, you're probably like, "Sounds like a good deal." Now you have a disgruntled person who owns a piece of your company and, even worse, the remaining workers are working for that ex-founder. They are building all the value in the company and the ex-founder who got fired is sitting there with all their shares going, “That’s right. Make it valuable.”

The take away here is avoid problems by paying yourself. Paying your payroll taxes and thinking of your co-founders wages like a marital pre-nup.
Kirsty: As the founders, you are going to need to hire employees. Much has been said in previous classes about how to find those people, about what makes a good fit, and about how to make them really productive employees. When you actually find somebody, how do you hire them? What's involved? Employment is governed by a huge raft of laws. Therefore, it's important to get this right. It's again the nitty gritty stuff that, as long as you know the basics, you can probably keep yourself out of most situations. As soon as things get complicated, you need to get yourself involved with a specialist. The first thing you need to do is figure out if the person is an employee or a contractor. There are subtle differences to this classification. this is important to get right because the IRS takes a big interest in this. If they think you got it wrong they will come after you with fines.

Both an employee and a contractor will require documents that assign any IP that they create to the company. That's obviously really important. The form of the document is very different for each type of person and the method of payments are very different. Generally a contractor will be able to set their own work hours and location and they will be given a project where there is an end result. How they actually get to that will not be set. They'll be using their own equipment and they won't really have any say in the day to day running of the company or the strategy going forward. A contractor will sign a consulting agreement. When the company pays them, the company doesn't hold any taxes on their behalf. That responsibility is on the individual. At the end of the year, the company will provide what's called a form 1099 to the individual and to the IRS, which they use to prepare their personal tax returns.

The opposite side of this is an employee. An employee will also sign some form of IP assignment agreement, but when the company pays them, the company will withhold taxes from their salary. The company is responsible for paying those taxes to the relevant state or federal authorities. At the end of the year the employee receives a W2 form, which will then be used to prepare their personal tax returns.

The founders need to be paid. So do employees. It isn't enough to just say, "Well, I am paying them in stock. That can be their compensation." They need to be paid at least minimum wage. In San Francisco, which has a slightly higher minimum wage than California as a whole, that works out to about two thousand dollars a month. It's not a huge amount but it can add up.

There's are other things that you need to make sure you have if you have employees. The first thing is that you're required to have workers compensation insurance, especially if you're in New York. The New York authorities that look after this will send really threatening letters saying, "You owe fifty thousand dollars in fines because your one employee that's being paid minimum wage has not paid the twenty dollars a month of workers compensation fees." It is really important that you set that up. The other thing that is important is that you need to see proof that the employee is authorized to work in the US.

Founders are not payroll experts and nobody expects you to be one. This is all about the basics. You absolutely must use a payroll service provider who will look after this for you. Services like Zen Payroll are focused on startups. They help you get this set up in the easiest way possible so you can go back and concentrate on what you do best. In the example that Carolyn gave just a few minutes ago, if that company had actually set themselves up with a payroll service provider, all of that heartache would have gone away because it would have been looked after for them. They were trying to save money by not doing it and look where it got them.

That's the key thing. Use a payroll service provider and make sure that you understand the basics of employment.
Carolynn: Somebody at YC once said, "You're not a real founder until you've had to fire somebody." Why is that? Because firing people is really hard. It's hard for a lot of reasons, including that founders tend to hire their friends. They tend to hire former co-workers or they get close to their employees because working at a startup is really intense. But in every company there's going to be an employee that doesn't work out and firing this employee makes a founder a real professional because he or she has to do what is right for the company instead of what is easy. Best practices for how to fire someone: number one, fire quickly. Don't let a bad employee linger. It's so easy to put off a difficult conversation but there is only downside to procrastination. If a toxic employee stays around too long, good employees may quit. If the employee is actually screwing up the job, you may lose business or users.

Number two, communicate effectively. Don't rationalize. Don't make excuses. Don't equivocate about why you are firing the employee. Make clear direct statements. Don't apologize. "We're letting you go," not, "I'm so sorry your sales didn't take off this quarter, blah, blah, blah." Fire the employee face to face and ideally with a third party present.

Number three, pay all wages and accrued vacation immediately. This is a legal requirement that we don't debate or negotiate.

Number four, cut off access to digital systems. Once an employee is out the door, cut off physical and digital access. Control information in the cloud. Change passwords. We had a situation at YC where one founder had access to the company's GitHub account and held the password hostage when his co-founders try to fire him.

Number five, if the terminated employee has any invested shares, the company should repurchase them right away. The takeaway here is that, surprising as this may sound, one of the hallmarks of a really effective founder is how well he or she handles employee terminations.

Kirsty: The basic tenant to all of this is keep it simple. Do all the standard stuff and keep it organized. Make sure you know what you're doing. Equity ownership is really important, so make sure you are thinking about the future rather than the three months of the history of the company. Stock doesn't buy itself, so make sure you do the paperwork.

Make sure you actually know about the financing documents that you're signing. It's not enough to just say, "I'll take your hundred K." You and the employees need to be paid. Everybody needs to assign IP to the company. If the company does not own that IP, there is no value in the company. If an employee must be fired, do it quickly and professionally.

We didn't mention knowing your key metrics. At any time you should know the cash position, you should know your burn rate, you should know when that cash is going to run out so you can talk to your investors. A lot of running a startup is following the rules and taking it seriously. It's not all the glamorous bits that we see in movies and TV shows. You do have to take that seriously.

Audience Member #1: How would you advise searching for an accountant and when in the process do you need one?

Kirsty: There are two different things. There is a book keeper and there is a CPA, an accountant. Generally book keepers will categorize all your expenses and CPAs will prepare your tax returns. In the very early days it's probably fine for the founders to just be able to see the bank statements and see those expenses coming out. Tax returns have to be prepared annually, so at some point in that first year of the company's life, some service is going to need to be engaged to do that. It's not worth the
founders time to do it. There are services available like inDinero which try to make things as effortless as possible from the founders's point of view. You do need to get a CPA at some point because you need to file your annual tax returns for the company.

**Audience Member #1:** How do you find one?

**Kirsty:** Finding one is tough. The best is through recommendations. With any kind of specialist, a CPA or an accountant or a lawyer, it's always best to use people who are used to dealing with startups. Not your aunt who lives in Minnesota and doesn't actually know how startups work.

**Audience Member #2:** All things considered, what should be my budget for incorporating, for the lawyer, for getting the deal to buy for my effort seed rounds? And then for hiring the first employees. How much money should I set aside for that?

**Carolynn:** In terms of incorporation, don't spend a dime on that. You can do that online. Well, actually it does cost a little bit. Incorporating online using a service like Clerky is inexpensive. In the hundreds, not in the thousands. You don't need a lawyer for that part. When you actually need to hire a lawyer depends on what business you are starting and how complicated it is. Do you have a lot of privacy policies, is HIPPA involved? You mentioned raising your seed round, how much money are you raising? Who are the investors? What kind of terms are in the term sheet? Sometimes that dictates whether or not you need to get legal counsel.

**Kirsty:** Services like Clerky can help if you are using very standard documents for the fundraising. There are very basic fundraising documents you can use that cost less than a hundred dollars, which can save you some legal fees.

**Audience Member #3:** Do you have any advice or comments on the complexity that comes with working with cryptocurrencies or cryptoequities?

**Kirsty:** Oh wow. That's a tough question to end with. Yes, there are some issues. Often banks struggle to deal with companies that are working with cryptocurrencies because they haven't quite figured out how to deal with that sort of thing yet. Generally a lot of it is very product specific.

**Sam:** Thank you very much!

**Carolynn:** You're welcome.
Lecture 18: How to Work with Lawyers at a Startup


I recently read a post over on VentureHacks titled, "Top Ten Reasons Entrepreneurs Hate Lawyers" written by Scott Walker (who blogs on legal issues for entrepreneurs). I know that people have an allergy to lawyers out of fear of being screwed. Much of this is unfounded – some is not. If you're a startup and you don't have a close relationship with a few law firms you're really missing one of the most important relationships that any entrepreneur can have.

If you don't follow the image reference above or the tag line, "You don't need double talk; you need Bob Loblaw" (try saying it out loud), and if you care! the link is here.

This all got me thinking about a post on how to best work with lawyers. This is stuff I tell people verbally at least twice / month so I'm glad to finally get it into written format.

1. When to get a lawyer - If you plan to be a venture or angel backed technology company (what I mostly write about) the best time to start meeting and getting to know lawyers is long before you ever start your company. Many people start companies arse backwards. They get together with a few buddies, bounce around ideas, build some code (sometimes internally, sometimes through contractors), start talking to potential angel investors and then register their company. This is one of the biggest source of future problems for the company. I write about some of the lessons in my post on Startup Mistakes.

So eventually you have your company funded but only 2 of the 5 people who started the company are still around. You never got around to agreeing exact equity splits but you had many conversations about it. I wonder how the shunned three are going to feel when you make your millions? Your memory – they never worked very hard and didn’t want to commit. Their memory – the idea was theirs in the first place and you ran with it and didn’t include them.

Your contractor wrote a great V1 of the product and it helped you get angel funding. Now you have money and a crack new tech architect and you'll have to rebuild everything. Shame about not getting it in legal writing that you owned the original IP. But I'm sure he'll never remember your company when Google buys you for $500 million and he's still eating Ramen – right?

Founded it as a California LLC but your potential VC wants a Delaware C-Corp? Forget to get around to setting up that Employee Stock Option Plan and want to be able to give the early guys their options at a low strike price? Shame about that pesky FAS 157 ruling. Oh, they didn't cover that in your Stanford CS course? Sorry. I’m sure the IRS will be flexible and your friends will forgive you for their big, unexpected tax burden.

2. But I don't have money to pay a fancy lawyer – I'll just have my cousin do it - Don’t. Your cousin specializes in entertainment law – that's different. I know he’s smart but you wouldn’t hire a Javascript developer to do your database design – would you? Here’s a hack for you. Most lawyers that work with startups are willing to work on a deferred payment schedule. They'll only do this if they believe you're a high potential team and are likely to raise money at some point. Consider it a sales & marketing expense for them. They want to lock in future clients at an early stage.

If you make this cut then they will usually defer the payment until your funding. For a very small fee they can get your Delaware C corp registration, make sure that you have IP protection and ensure you didn't make an early bumbling mistakes that you'll pay for dearly in the next 7-10 years of hard work. They’ll help you ensure that any money you raise doesn't come with surprise terms that are hidden in legalese. They usually ask for warrants (basically like a stock option) in exchange for taking a deferred fee.

Like anything in life, if you want a fair deal on the deferred fee and warrant percentage you need to talk to a few lawyers to make it competitive. They'll find a way to get to the right price if they believe you’re high potential.

3. Additional value outside of legal documents – Most lawyers who work with early stage startups secretly want to be entrepreneurs but they earn too much money to quit – the golden handcuffs. I’ve found that the best of them think like entrepreneurs, though, and hang out at startup events. So you can become friends with these people just like NORMAL people and, don’t worry, you’re not going to get a bill for having a beer with them.

I find that one of my best sources of deal flow is from lawyers. Because many great entrepreneurs work with lawyers in registering their companies they have their ear to the pavement on the earliest of company formations. I recently wrote a post on how to get access to VCs and angels where I outlined why lawyers are such an important tool for VC introductions.

Think about it – on every company we fund there is a lawyer representing the company. We spend hours with them discussing and negotiating the details of the company. When we invest they are often the company counsel so we see them at board meetings. When we want to sell or IPO companies they're there again. Our lives are intertwined.

So I tell you this because if you build a strong relationship with an early-stage VC they can be a great source to help you with your funding. They can and will provide introductions. The best firms have whole departments dedicated to just this. I know that DLA Piper does. I think they're one of the best firms nationally at this. The website...
for their Venture Pipeline group is here. I'm sure other firms have similar – feel free to add in the comments section.

4. How to choose a lawyer

- First, you need to pick a firm. I personally like startups to work with firms based on three criteria: local, right sized and startup focused.

Local: many entrepreneurs like to use their old friend from Boston even though they’re now living in Boulder. That makes no sense to me. The additional benefits of working with a local law firm or the local branch of a national law firm are too big to ignore. As I said previously they’ll tap you into the funding source. They’ll invite you out to events in which you’ll meet their other clients, you can get to know them socially and hopefully develop a real mentorship relationship where every conversation is not on the clock.

Right sized: I tried to work with one of the best known firms in Silicon Valley. In the initial stages I found that I struggled to get their attention because they also represented all of the biggest titans in town. My piddly little funding deal was dwarfed in comparison to the huge M&A deals they were negotiating. Sure, they’ll take you on. But try getting the time of the A players. You’ll get the scraps. I think you’ll end up with a newer lawyer who’s trying to build her book of businesses. And worst still, you won’t even be her top priority. You’ll also suffer from their billing rates. I learned all this the hard way. Every town has firms that focus on startups – find them.

Startup Focused – My one carve out for working with the big brands is that many of them have internal practices that focus on startups. I already mentioned DLA Piper. I’ve met several people from Cooley Godward who have stellar reputations in this category. I know Mike Lincoln in Washington DC does. I worked with Sam Angus at Fenwick & West. He’s totally tapped into the startup communities in Silicon Valley and a bit in LA. So it doesn’t have to be a small firm.

But as with consulting, PR, web design and even VC – it’s not just the firm it’s also the individual. In every firm there are A, B and C players. Good people and evil people. Focus on the partner you would be working with.

5. How to manage costs

- One of the biggest frustrations that people have with lawyers are unexpected costs. You have this perception that they’re billing you for the lunch meeting they invited you to and the friendly banter emails flying back and forth. I know the industry as a whole has been plagued with criticism over over billing and personally I think much of this criticism is deserved. I saw the same bad behavior in consulting.

But it doesn’t have to be that way and the responsibility lies with you. For company registration, angel deals, Series A & B funding, Employee Stock Option Plans (ESOP), IP filings and even litigation it doesn’t need to be that way. I always try operate on the “Fixed Fee +” arrangement. If you’re doing a “vanilla” funding, the lawyer pretty much knows how much effort will be required. So I ask them for a “fixed” budget. That’s the amount that at the end of the project I expect to see. The “+” is what I’ll allow them to bill me for in an “exception” if (and only if) something totally whacky pops out. And I ask them to tell me in advance if it bubbles up.

The key to being happy with your bill at the end is simple: no surprises. Be fair, let your lawyers earn. But make it competitive. I also am a very loyal person. I always tell the lawyers that I work with that if they’re fair on pricing and do great work I promise to make it up to them in intros for future work. I think most people would say I’ve held that bargain.

Also, don’t send shit to your lawyers that you can do yourself. I’ve seen startup CEO’s send requests like, “can you please update our Cap Table with the latest stock option allocations approved at the board meeting?” My email response (in my head, not sent), “can you please waste more money having expensive lawyers do something that you could / should easily do for free?” Be practical about what you ask your lawyers to do.

6. Traps to look out for – Bram Cohen, the investor of BitTorrent, covered the topic on VentureHacks here. One issue he talked about was working with partners. I also like to work with partners. It’s true that you get a higher billing rate but as Bram points out they can often get the work done in way less hours. If they’ll agree to do your work directly – go for it.

But I also know it’s not realistic for the partners to do all of the work. So make sure when you do your analysis on which firms to work with that you also meet the associates you’ll actually be doing work with. I work a lot with Dave Young(david.young@dlapiper.com) at DLA Piper. He works a lot with Nick Hobson, an associate who’s a star. Associates matter. Get to know them. [does that qualify for the discount on my last engagement with you guys ] I like the WYSIWYG approach to working with lawyers – I don’t want the partners pitching the work and it gets completed behind the scene by somebody I’ve never met.

Another big “gotcha” for me is that you expect lawyers to help you negotiate good deals. What I found is that most lawyers will tell you what all the terms mean and sometimes will tell you what is commercially normal but they NEVER explain to you just how certain terms can be used to screw you in the future. You cannot just say these clauses are “legalese” and I’ll let my lawyer figure them out. You need to own your legal agreements. You need to know how liquidations preferences work. You need to know how “tag along” rights could potentially screw you. You can’t just understand “redemption rights” in theory. That’s why I love VentureHacks. And Brad Feld. He was the first guy that I saw demystify the legal terms in his term sheet series. This is a must read for all entrepreneurs.
When you’re doing biz dev deals or customer deals you need to start with a list of the things that you think could go wrong or that you want to protect against. Write them in English (or your native language) and then ask your lawyer to think about how to protect for them. Not the other way around. One CEO I work with was working on a agreement with a major company. The agreement they sent said, “we may offer your services to our customers.” This went unnoticed by the CEO and the lawyer. I asked a simple question. WTF is the definition of “may?” In human speak that sounds like they also have the other option, “may not.” What a waste to sign an agreement that someone “may” offer your service. That kind of shite doesn’t always get picked up by lawyers.

Be detail oriented. Own your outcomes. Lawyers are your support staff not your brain.

7. The good guys – There are many. Giff Constable has started a great national list here. It’s an awesome service to entrepreneurs, thank you. True Ventures goes one step further with a broader recommendation list here (I hope to add this tab to our website in our next rev. True is always one step ahead of the curve) My additions below:

- I’ve already covered David Young at DLA Piper. He mostly covers Southern California.
- I really had a great experience with Sam Angus at Fenwick & West when I lived in Silicon Valley.
- I have worked extensively with Scott Alderton (yes, he was having a bad hair day in the picture on the link!) of Stubb Alderton, a firm that focused exclusively on early-stage tech, media and game companies in Southern California. They have a stellar reputation and know how to work with the earliest of starts with entrepreneurs.
- If you’re doing litigation work and want one of the best national firms who will work on a contingency arrangement (e.g. much of their fees are paid only if they win) then check out Steven Sklaver over at Sussman Godfrey. Disclosure: he’s my cousin. And he’s also a Cowboys fan. But if you can stomach that he’s a star and contingency work/risk sharing on IP claims is key!
- Don Lee at Buchalter Nemer was referred to me by Todd Gitlin and came on very high recommendations as an early-stage, true entrepreneur’s perfect lawyer because he understands how to work with startups.
- On later stage LA Tech M&A deals I’ve heard there’s none better than David Hernand of Gibson Dunn
- In Silicon Valley the name I hear mentioned most often for very early stage deals is Joey Tran from Fortis. He represented us at one firm I invested in and did a great job.
- We’ve had great experiences with Michael Pfau in Santa Barbara.
- I never met a person who didn’t talk about Mike Lincoln of Cooley Godward in DC.

Anyone who does IP law in Southern California seems to use Knobbe Martens.

Feel free to add any of the good guys in the comments and/or make sure to them over on Giff’s list. Feel free to also add your best tips (or traps) on working lawyers.

Lecture 18: Useful resources

http://www.clerky.com/
http://www.zenpayroll.com/
http://www.zenefits.com/
Lecture 18: Top 10 reasons why entrepreneurs hate lawyers

http://venturehacks.com/articles/hate-lawyers

Thanks to Walker Corporate Law Group, a boutique law firm specializing in the representation of entrepreneurs, for supporting Venture Hacks this month. This post is by Scott Edward Walker, the firm’s founder and CEO. If you like it, check out Scott’s blog and tweets @ScottEdWalker. He’s also writing a new series on VentureBeat: Ask the attorney – Nivi

Last week I offered 5 New Year’s resolutions for closing deals in 2010. This week, I thought I’d have a little fun and address the issue of entrepreneurs’ frustration with lawyers.

A recent tweet from Bram Cohen, the inventor of BitTorrent, captures this frustration well: “Lawyers are like phone companies. Their bread and butter is in tricking you into racking up minutes.”

There’s a time in just about every entrepreneur’s career when he or she has wanted, in the words of Shakespeare, to “kill all the lawyers”. In the spirit of David Letterman, here are my Top 10 reasons entrepreneurs hate lawyers (I should point out that “hate” is too strong a word to describe the feelings of most entrepreneurs, but it makes for a catcher title than “dislike” or “complain about”). Click here for a brief video version of this post.

#10 – “Because they don’t communicate clearly or concisely”

Lawyers love speaking legalese and hearing themselves talk. I learned this first-hand as a corporate associate for nearly eight years at two large New York City firms. The tax lawyers, the employee benefits lawyers, the antitrust lawyers and the rest all spoke their own language. As a corporate associate in charge of quarterbacking transactions, I dealt with the various legal specialists and had to learn their mumbo jumbo. At times, I was as frustrated as the clients.

In the book Garner on Language and Writing, Former U.S. Solicitor General Theodore Olsen wrote, “Legalese is jargon. All professions have it. All professions use it as a substitute for thinking, and they all use it in a way that makes them appear to be superior. Actually, they appear to be buffoons for using it. The legal profession may be the worst of all professions in using jargon. It’s not necessary to communicate that way. You’re really not communicating, and you’re not really thinking.”

#9 – “Because they don’t keep me informed”

Lawyers often keep their clients in the dark. The real estate lawyer I hired to handle the sale of a property came highly recommended and seemed like a good guy. But I never knew what was happening throughout the process. I showed up to the scheduled closing only to learn it was postponed because of some wrinkles, including the buyer’s financing.

Tom Kane, a legal consultant, notes: “[A] failure to communicate often (as in constantly, frequently, persistently, regularly…) is not only foolish from a professional standpoint (as in discipline by the bar, keeping professional insurance premiums reasonable, and so forth), BUT it is just dumb marketing. One could even say it is marketing malpractice.”

#8 – “Because they are constantly over-lawyering”

Corporate lawyers often have a one-size-fits-all approach to deals. I recently represented a software company in a relatively small business sale (about $10 million). The buyer was represented by a large law firm that sent an acquisition agreement with three pages of environmental representations. When I explained that none of the environmental reps (or indemnities) was applicable to the target because it was a software company with one office lease, the corporate counsel got on a soapbox about his client “not assuming any environmental risks.” He even patched in the firm’s environmental lawyer to support his argument.

As John Derrick, a California appeals specialist, points out in his book Boo to Billable Hours, “Just as the cost-plus contractor has no financial incentive to keep the price down once hired for the job, so the lawyer who charges by the hour has little incentive — at least in the short term — to keep down the hours billed. To the contrary, the lawyer’s incentive is to bill as much as possible. The result can be unnecessary lawyering.”

#7 – “Because they have poor listening skills”

While lawyers love hearing themselves talk, they are often not very good at listening. Entrepreneurs want their lawyers to listen carefully to their concerns and address them appropriately; and they don’t want to be interrupted. I feel the same way, particularly when I am negotiating a transaction and trying to close a deal. I have sat in too many conference rooms negotiating with other lawyers as they played with their Blackberries and answered calls on their cell phones. This is not only rude, but it’s also bad lawyering.

From the Wabet Blog: “While great corporate lawyers have several different attributes, one stands apart from the rest: being an exceptional listener. First of all, it’s essential that the corporate lawyer is always ready and able to listen to the client’s description of [his or her] goals and needs. This sounds trite, but involves a set of skills that is more than simply hearing the words spoken or reading the words on the written page. The exceptional corporate lawyer looks beyond the words to delve into the facts, circumstances and other aspects that define the situation… Some of the skill is derived from training, but to a large extent the exceptional corporate lawyer applies his or her experience and the wisdom derived from that experience.”

http://startupclass.samaltman.com/ (gekregen van Inne ten Have innen darwine.nl 06-42804208) 376 / 412
#6 – “Because inexperienced lawyers are doing most of the work”

This is the dirty little secret at most law firms, particularly large ones. It even has a name: “leverage”. Law firms try to create the highest possible ratio of associates to partners. The higher the ratio, the more money the partners make. For most entrepreneurs, this generally means paying for the training of young associates.

I discuss this issue in my blog post Behind the Big Law-Firm Curtain: The Good, The Bad, The Ugly. “The reality is that the smaller the client — the smaller the transaction — the further down the ladder the work gets pushed at the big law firms. That’s the way these firms work. The entrepreneur may meet the senior partner at the first meeting for his $15 million acquisition or $3 million financing, but that partner then goes back to his office, calls the assigning partner and gets some young associate to start cranking out the work.”

#5 – “Because they spend too much time on insignificant issues”

Lawyers are notorious for failing to prioritize issues. This is especially true in small transactions. Since I moved to Los Angeles from New York City in 2005, I have handled predominately middle-market M&A transactions, financings and restructurings, a departure from the billion-dollar deals I handled in New York. I expected lawyers on these transactions to produce documents relatively quickly and focus on the key issues of a deal, particularly in venture capital transactions that benefit from standardized documents from the National Venture Capital Association. Instead, I found much of what I found in New York: lawyers spending needless time fighting over insignificant issues.

Foundry Group co-founder and managing director Jason Mendelson recently asked, “Why can’t lawyers know when to leave well enough alone and not feel like every piece of paper needs a mark up? Especially given how expensive lawyers are these days, why on earth would the culture of ‘must mark up documents to show value’ persist? (Answer: lawyers make more money). Especially in the world of venture financing, this is very frustrating.”

#4 – “Because they don’t genuinely care about me or my matter”

Too few lawyers are passionate about the practice of law. Before launching my own firm, I worked alongside many big-firm lawyers who didn’t seem to enjoy what they were doing. This translates to indifference toward clients.

This quote from Zappos CEO Tony Hsieh in a recent New York Times interview struck a chord with me: “I just didn’t look forward to going to the office. The passion and excitement were no longer there. That’s kind of a weird feeling for me because this was a company I co-founded, and if I was feeling that way, how must the other employees feel? That’s actually why we ended up selling the company.”

That’s how I felt at the law firms where I worked. There were a number of passionate superstars at each of my previous firms. But many others were burned out and just going through the motions. “Just another fuck’n deal,” one of my former colleagues once complained to me. That’s why I launched my own firm: to create a team of passionate, hard-working corporate lawyers who love what they do and love helping entrepreneurs.

#3 – “Because their fees are through the roof”

As I discuss in the introductory video on the home page of our website, the traditional law firm business model is broken. Legal fees have sky-rocketed over the past decade, with lawyers at some national firms billing more than $1,000 per hour and lawyers at smaller, so-called “regional” firms, billing more than $600 per hour (see “Law Firm Fees Defy Gravity, Annual Survey Shows”). The number one thing driving these outrageous rates: overhead. Traditional law firms simply pass huge overhead costs onto their clients — expensive office space with lavish artwork and dramatic views; large support staffs complete with librarians, and receptionists; and, of course, high-paid associates.

As a result of the recession and this broken business model, large law firms have recently shed associates in large numbers. LawShucks reports, “2009 will go down as the worst year ever for law-firm layoffs. More people were laid off by more firms than had been reported for all previous years combined.” But as Dan Slater argues in his recent New York Times DealBook post, Another View: In Praise of Law Firm Layoffs, “These layoffs — which in many cases have been paired with salary freezes or cuts and significant reductions in law school recruiting — are the best thing to happen to the legal industry in years. Call it a blessing amid recession. Start with the benefit to cost-conscious corporate counsel, who for too long have been bilked by a law firm compensation model that leads lawyers to prioritize their ‘hourly quotas,’ which determine year-end bonuses, over quality service.”

#2 – “Because they are unresponsive”

We’re all busy, but that’s not a viable excuse for failing to promptly return a client’s phone call or email. Clients may have differing definitions of “promptly,” but one business day is a good starting point. I experienced unresponsive lawyers as a client in personal matters, and I experience it as a corporate lawyer trying to close deals on behalf of my clients. Entrepreneurs crave immediacy (and so do I).

A recent deal I was on ran days late, requiring an all-hands conference call to finalize a few key issues in the acquisition agreement. I distributed an updated version the same day with instructions to the lawyer on the other side to call me for an update before he left for the weekend. The weekend
passed. I heard back from the lawyer on Monday afternoon, over email — and he had sent a new blacklined version with all new issues raised.

#1 – “Because they are deal-killers”

Lawyers are often viewed as deal-killers because of their failure to set a positive tone and their annoying habit of raising all sorts of reasons why a particular deal won’t close or why a particular idea won’t work. One of the better lawyers I worked with at a firm often said: “Good lawyers are able to identify significant potential legal problems; great lawyers provide solutions to those problems.”

As James Freund, a professor and retired partner at Skadden Arps in New York, points out, “In a transactional practice, nothing comes easy. There are invariably two opposing points of view on significant issues, and the parties will even clash… over a circumstance that may never come to pass. Every disputed issue has to be resolved in order for the deal to take place. And the business lawyers bear the primary responsibility for getting it done. Viewed in its broader context, this activity falls under the rubric of problem solving. Unless you’re a problem solver, you’re unlikely to be an effective business lawyer. And the problems that stand in your way aren’t limited to transactional matters… they can involve dealings with regulatory agencies, tax planning, strategizing about how to protect intellectual property, and on and on.”

Conclusion

While much of this list includes criticisms of my industry, I hope it helps initiate dialogue among entrepreneurs and the lawyers who represent them, to improve the value of the services we offer. And, please remember, I put this list together in the spirit of having a little fun. What experiences have you had with lawyers? Feel free to share in the comments section.

If you like this post, check out Scott’s blog and tweets @ScottEdWalker. He’s also writing a new series on VentureBeat: Ask the attorney. If you want an intro to Scott, send me an email. I’ll put you in touch if there’s a fit. Finally, contact me if you’re interested in supporting Venture Hacks. Thanks. – Nivi

Lecture 18: Startup Company Lawyer

http://www.startupcompanylawyer.com/

Yorim Taku

Below is a comprehensive list of posts.

**INCORPORATION**

1. What do you need to do before you quit your job to form a startup company?
2. When do I need to incorporate a company?
3. What type of entity should I form?
4. What state should I incorporate in?
5. What is a certificate of incorporation?
6. What are bylaws?
7. Why should a company follow corporate formalities?
8. What does the board of directors do?
9. What liabilities do the board or officers face?
10. What do stockholders need to approve?
11. How many shares should be authorized in the certificate of incorporation?
12. What does fully-diluted capitalization mean?
13. What is par value?
14. What is an agent for service of process?
15. How do you calculate Delaware franchise taxes?
16. What trademark and other legal issues are involved in selecting a company name?
17. What inspection and information rights does a stockholder have?

**FOUNDERS ISSUES**

1. Should founders pay for their stock in cash or contribute intellectual property?
2. Should founders stock be subject to vesting before a venture financing?
3. What should the vesting terms of founder stock be before a venture financing?
4. What is an 83(b) election?
5. What is qualified small business stock?
6. Obama proposes no capital gains tax on qualified small business stock
7. What is Series FF stock?
8. What is Class F common stock?
9. What is TheFunded Founder Institute?

**STOCK OPTIONS/COMPENSATION**

1. What is Section 409A?
2. How do you set the exercise price of stock options to avoid Section 409A issues?
3. What’s the difference between an ISO and an NSO?
4. Should a company allow early exercise of stock options?
5. Should a company allow exercises of stock options with promissory notes?
6. How many shares should be reserved under a stock option plan?
7. How much should you pay an executive in a startup company?

GENERAL
1. What is important in a confidentiality agreement or non-disclosure agreement (NDA)?
2. What are securities laws?
3. What is an accredited investor?
4. Can a California company have unpaid interns?
5. Is crowdfunding legal?

CONVERTIBLE NOTE BRIDGE FINANCINGS
1. Should a startup company raise its seed round using a convertible note or Series A Preferred Stock?
2. What does a convertible note bridge financing term sheet look like?
3. What should the interest rate for a convertible note be?
4. What should the conversion discount be for a bridge note?
5. What does a convertible note bridge financing term sheet look like?
6. What type of financing forces an automatic conversion of the promissory note into Preferred Stock?
7. What happens to the convertible promissory note if the maturity date is reached and there hasn’t been a financing?
8. What happens if the Company is sold after the convertible note is issued and before the maturity date or the next round of financing?
9. What should the maturity date of the convertible note be?
10. What should the terms of bridge loan warrant coverage be?
11. Can you have multiple closings in a convertible note bridge financing?
12. What does subordination mean in a convertible bridge note?
13. What is a security interest in connection with a convertible note?
14. What should the representation and warranties in the note purchase agreement be?
15. Why should a majority of investors be able to amend the convertible notes?
16. Who pays legal fees in a convertible note bridge financing and how much does it cost?
17. What is the economic difference between a conversion discount and warrant coverage for a convertible note?
18. What is the California Finance Lenders Law?
19. Is convertible debt with a price cap really the best financing structure?
20. What are the terms of Yuri Milner/SV Angel's Start Fund $150K investment into Y Combinator companies?
21. What is convertible equity (or a convertible security)?

SERIES A FINANCINGS
1. What trends does WSGR see in venture financings?
2. What does a Series A term sheet look like?
3. What do definitive agreements for a Series A financing look like?
4. Are Series A term sheets binding?
5. What is preferred stock and why is it issued to investors?
6. How do you calculate Series A price per share?
7. Can you have multiple closings in a Series A financing?
8. What is a dividend preference?
9. What is a liquidation preference?
10. What is the amount of a typical liquidation preference?
11. What is the difference between non-participating preferred stock and participating preferred stock?
12. What is a cap on a participating preferred liquidation preference?
13. What is the priority of the liquidation preference when the Series B financing occurs?
14. Why do preferred stockholders have odd economic incentives upon a sale of company when they have non-participating preferred stock or participating preferred stock with a cap?
15. What are redemption rights?
16. Why is preferred stock convertible into common stock?
17. When should preferred stock be automatically converted into common stock?
18. What is anti-dilution protection?
19. What is weighted average anti-dilution protection?
20. What is full ratchet anti-dilution protection?
21. What is a pay to play provision?
22. What are carveouts to anti-dilution protection?
23. What are protective provisions?
24. What stockholder approval is necessary to complete a venture financing?
25. What are information rights?
26. What are registration rights?
27. What are demand registration rights?
28. What are piggyback registration rights?
29. What are S-3 registration rights?
30. What is a market standoff or IPO lockup provision?
31. What is a right of first offer or right to maintain proportionate ownership in future financings?
32. What is a right of first refusal and co-sale agreement?
33. What is a drag-along or bring-along provision?
34. What should the composition of the board be like and how are the board seats allocated?
35. What are board observer rights?
36. What is a management rights letter?
37. What should legal fees for a Series A financing be?
38. What should the terms of the no shop be?
39. Why should a term sheet be confidential?
40. What are the conditions to closing of a Series A financing?
41. What does the legal opinion cover?
42. What is Form D and what information gets publicly disclosed to the SEC in a financing?
43. How do the sample Y Combinator Series AA financing documents differ from typical Series A financing documents (or what’s the difference between seed and venture financing terms)?
44. How do the sample Series Seed financing documents differ from typical Series A financing documents?
45. What is upgradeable Series A preferred stock?
46. How do you find federal and state government funding opportunities for clean tech and other companies?

SERIES B FINANCINGS

1. What does pari passu mean?
2. Should the Series B liquidation preference be senior to the Series A?
3. Should the Series B have a separate protective provision?

DOWN ROUNDS

1. What deal terms appear in down round and highly dilutive financings?
2. How can a board decrease litigation risk in an insider-led down round or dilutive financing?
3. If a down round financing is led by a new outside investor, does the board need to be concerned by the business judgment rule?
4. Should board members representing inside investors vote on the board resolutions authorizing the insider-led down round?

M&A

1. What stockholder approval is necessary to sell a company?
2. What are directors’ duties and what can they do to protect themselves in a sale of company?
3. What does a liquidation preference spreadsheet look like?
19. How to pitch
Lecture 19: Sales and Marketing

http://startupclass.samaltman.com/courses/lec19/

Tyler Bosmeny

My name's Tyler. I'm the CEO of Clever. What I want to talk today is about sales. I have a little insight into this. I graduated college. I actually studied math and statistics, like some of you here in this room. I thought I was destined for this world of finance. I was about to go start at a hedge fund, but at the last second a friend of mine roped me into joining his startup to do sales, which I knew nothing about. I had to figure it out on the fly and spent a couple of years there figuring out sales for this very early stage company. When it came time to start Clever, I started Clever with two co-founders who were very technical and one very product oriented. We wanted to build this product for schools and I thought that experience would have no relevancy whatsoever. It turns out that some of the things that I picked up at this previous job where I was figuring out sales have been HUGE parts of what's made Clever grow so quickly today.

Quick background on Clever. We build software for schools. We are an app platform used by developers and it's used today by about one and five schools in America. And we started it about two years ago.

And so sales has been a big key piece of that. And I want to use this time to just share some of the things that have worked for me along the way. Of course there's a million ways to do this. So you'll find what works for you.

So first I want to start about most - how I used to perceive sales. And a lot of people see sales as having this, you know, a lot of mystique around it, you know. It's people who are, you know really articulate and impossibly charming. And they have these, you know, killer closing lines that they use. And I think - this how I saw sales. And think this is how a lot of founders I talk to see sales because they say things to me like, "You know we're just going to work on the product and build a great product and then when it's finally finished, we're going hire the sales people." And what I've learned is that hire the sales people - as a founder the reality is that's you. And so you know Paul Graham likes to talk about how there's two things he should be doing at any point and time when you're starting your company. You either talking to your users or you're building your product. And that talking to your user's part, that's selling. And so you know this is intimidating some people 'cause there like, "I've never done sales. And I wouldn't even know where to begin." But it turns out that as a founder you have some unique advantages that make it possible for you to be really, really good at sales. And one of those is your passion for the product and what you're building. And the second is your industry knowledge of what you're, of the industry and the problem that you're solving. And those two things actually totally trump sales experience from what I've seen.

So this is actually my co-founder doings sales. This is what sales looks like, you know in the very early stage of a startup. It's not done drapers. It's a lot of calls like these. But this is something that even as a founder who has never done it before, it is very easy to do but you have to commit yourself. And what we did at Clever is we dedicated one founder which was me to peel off and say, "Ok, Tyler you gotta go figure this out. And work on this full time because it's so important to our business."

So, a couple things that I've picked up about sales along the way in trying to figure this out. You know the first thing that everybody knows about sales is that say, "Ok. It's a funnel." And you have these different stages of funnels - of, of the funnel and you move your customers through it. Pretty common category is there's this prospecting category. We were trying to figure out who's even interested. Then
you're having a lot of conversations with is the second level of the funnel. Then you're finding out who's really serious and you want to close them and sign the deal. And then of course you're in the promise land of revenue. And what I thought would be interesting would be to talk about each, each of the stage. A couple of strategies that we've used at Clever that have worked really well. So that these aren't abstract things but things that you can hopefully use at your start up.

So prospecting. So, prospecting is the process of figuring out who will even take your call. And you know one of the things that I realized early on - So there's this guy at Everett Rogers who, who's created this technology life cycle adoption curve. And he describes it's as a bell curve. Where you've got, you know, your innovators who will try new things. And you've got your early adopters. Your mid-stage adopters. Your late adopters. Your lagers. And one of the things that was really helpful for me at understanding sales at an early start up is he's quantified the tale of this bell curve. And this part over here innovators - those are your potential customers. And it might seem discouraging that only 2.5 percent of companies are your potential customers or would even consider buying from a startup that has no users and no revenue. But actually, I found just the opposite. I found it to be extremely helpful that have this frame of mind because you realize when only 2.5 percent of companies will even take your call or consider using your product, you realize what a numbers game this becomes. So if you want to each that 2.5 percent and you want to get some early sales, you, you've - If you starting to do the math, you're hopefully starting to realize you have to do a lot of calling. You have to talk to a lot of people.

So early on in the early days Clever, this was my job. In the, in the two months - the first two months of YC I reached out to over 400 companies trying to to get them to take a call and talk to us about what we were building.

There's three ways that I have found most successful in prospecting and getting these people. One is your personal network. That's obvious. I'm not going to spend any time there. Another one is conferences. Which is surprising to a lot of people. And then the one that people are most familiar with is cold email. And when I say conferences, this what people think. They think I am talking about CES or you know, E3 or something. And actually the kind of conferences where sales happen look more like this. And we've got - we would in the early days would go to a lot of these because you've got to go to where your users are. And if there - if you're selling to CIO's and there happens to be a gathering of them at a hotel in Milwaukee - Guess what? That's where you should be. So we go to conferences like these. We get the attendee list in advance. We'd email every single person in advance and try set up meetings so when we get there every single minute of that trip was was well spent. It was huge in Clever's early days. This is where met all of our earliest customers.

The second thing I mentioned is cold email. A lot of people don't know how to write cold emails. It's actually really easy and the key is not to write a lot. Should be really concise. This is an email template that I used early on and you're welcome to copy it but it's really short. Here's who I am. Here's what I'm building and I'd love to talk to you about this. Could we find time tomorrow? It's really easy and you can customize this and find out for every business you want to sale to. Who's the right person to send it to? And you can send out quite a few of these.

So alright. That's prospecting and the reason this so important 'cause you got to build that first layer of the funnel.

Then you get them to take your call. And this is another place where a lot of founders I think just have a lot of questions about what to actually do. And the biggest thing to take away - In fact if you ONLY take away one thing from this presentation today - the number one thing you should remember is when you get them on the phone, remember to shut up. And that's really surprising to people. So many
founders when I helped them with their first sales pitch - they'd finally get somebody on the phone who wants to talk to them about their product and they're so proud of this thing that they've been building for the last three months that all they want to do is get on the phone and talk about every feature and talk about all the different things they can do. And talk about why it's the greatest thing in the world. And I have that temptation too. It's just part of being really proud of something. But it turns out that if you watch the best sales people - like the best sales people in the world - the top one percent. And you have a chance to listen in on a call with some of those people like I have - the most surprising thing is how little talking they do. In fact I've seen calls where the, where the sales person told me their goal was to only spend 30 percent of the call talking and have 70 percent of the call the other person. And they would ask a lot of questions. They'd say things like, "Why did you even agree to take my call today?" "This problem that we're talking about solving for you - how do you solve it today?" "What would your ideal solution look like?" And they're not doing to the talking. They're finding - they're doing everything they can to find out what this person needs and hopefully understand their problem even better than they do. That's what really great sales is. And in fact this is something I drill into everybody at Clever. It's really important part of sales and there's actually now if any of you use UberConference, they have this amazing feature where when you hang up a call it sends you an email automatically and tells you how much you talked versus how much the other person talked and looking at one of those emails - you know someone doing sales at Clever I get one of those emails - I can tell immediately how likely the sale is based on how much talking we were doing. So you got all these people. Now you got on the phone do a lot of listening. Really understand their problem.

And then the other part of this stage that surprises a lot of people is follow up. So here's a lot of different steps that you can imagine going through, you know. Emailing somebody. Not getting a response and emailing them back. Calling them. Leaving a voice mail. Having a pricing call. You know, there's probably like, you know sixty things up here on this slide that could be steps for closing a deal. These actually aren't just random things - this is one deal that this was a second deal Clever every signed. These are all the different steps that we had in order to get this done. And you can see there's a lot of really embarrassing things up there. Like I emailed somebody and they didn't respond. And I emailed them again and they didn't respond. And then I emailed them again. And this was from somebody who wanted to buy our product. Isn't that crazy? And that surprises a lot of people. I see so many founders who they have a great call with someone. They send an email. They don't hear back and they say oh that person might not be interested. Well guess what? This is what it looks like in the best case. And you really have to have kind this unhuman/ unreasonable willingness to follow up and drive things to closure.

Now I qualify with that with one thing which is to say when starting a company your time is EXTREMELY valuable. 'Cause it's your only resource. And you couldn't possibly do this for every single person who might buy your product. So your goal should be to get people to a yes or no as quickly as you can. Where you die is if you have a thousand maybes and sometimes I talk to founders who say, "Oh yeah I have this great pipeline of you know a hundred people who are - who have expressed interest in our product." And the maybes are what kill ya. If you can get to a yes or a no - In some ways a no is even better than a maybe because it allows you to move on and focus somebody who might be a yes.

So have this super human level of follow and ambition. But make sure you're focusing it on the right pieces.

Alright, so you've talk to someone - You've talked to a ton of people. You've had all these phone calls. You've followed up with them ridiculously to the point where they, they just know you're not going away and they've got to sign an agreement. This final step is something if you haven't done before it might seem all ope but it's actually really simple. It's called red lining. And so you'll send over an agreement. Their lawyers will mark it up. Your lawyers will mark it up and you kind of go back and forth. If you're
part of YC this is really easy because YC has standard template agreements that they give you so you don't have to find these and you can just, you know, use those. But they've never been available - you know if you weren't part of YC you kind of had to figure this out on your own. One of the things that I am really excited about is as part of this presentation YC has agreed to open source their deal documents. So these documents that YC founders use to get are not going to be available to everybody. So this should never - hopefully never be a barrier to anyone who wants to do sales for their start up. You've got some great documents. And the other thing I'll say about this a place where so many found - smart, smart people go wrong is you got to remember what your goal is. Your goal is to sign some deals and get some reference customers and get some validation and get some revenue. That - if you don't do that your startup is, is you know toast.

So in light of that it's really surprising how many smart people will want to do ten rounds of document review crippling over the most minor points because of pride - because of intelligence. Whatever. You know make sure the agreement is the way you want it but then sign and move on. I've seen founders spend month quivering over some indemnification clauses and their business would have been way better off if they'd you know just signed the deal and moved on to the next one. So that's one closing trap you can fall into.

I have two more. One other closing trap that I see founders struggle with a lot is they're talking to a company who says, "I will use your product but I just need one more feature." Or they say, "You know I'd love to use your product but it doesn't have this one feature. So we're just not ready." And to most people especially if you're ambitious, when somebody says that to you what you want to hear is, "Oh. I can build that feature. Great! You know I'll build that feature and then they're going to use my product." But the problem is it almost never works that way. In fact somebody telling you that the would use - they want to use your product but it's just missing this one feature - I would almost map that to a pass in your mind. Because nine times out of ten if you actually build that feature you go back to them and then there would be one more feature. Or they'd be some other reason that they're not using the product.

So if somebody says to you, "Hey. I want buy - you know there's this one thing that's preventing us from using your product." I would do one of two things. One say, "Well that's great! Let's sign an agreement and we'll put in the agreement that we're going to build this feature." In which case, you know, you know if you build it you're off to the races. Or more commonly, at what we did at Clever was we would say, "That's great. We're going to wait to see if we hear that demand from more customers." And then once you have a lot of customers requesting it, then you should build it regardless and then you're not - you don't have to worry about doing something that's customer one off. Which is what you really want to avoid.

So, don't fall into this trap. It happens all the time. And the other trap I would highly, highly recommend you try to avoid is the free trial trap. Because this happens all the time. People you know they go down this path with a customer. It seems really exciting and then the customer says, " Oh well - can I get a free trial?" And you can't blame them that's a totally reasonable thing to ask for. But the problem is when you are starting a startup you need revenue. You need validation. You need users. You need commitment. And free trials get you none of those things. So go - you do all this work and if you end up with a free trial, unfortunately you haven't made much progress as you think - it's actually terrible. You, you think you've made progress but really at the end of the free trial you're going to have to sale them all over again. So the way I handle this and has worked really well is when somebody says can I get a free trial, you say, "Well we don't - We don't do free trials but what we can do is we're going. We do annual agreements here. And what will do is for the first 30 or 60 days and for any reason you're not happy you can opt out." And that's a way to get you the things that you need while giving them
the comfort that they might need to take a chance on a startup. So that minor change is actually makes a night and day difference when you're thinking about these things.

Alright, so you've prospected. You've had a lot of conversations. Now you've closed people. You've gone through the red line process. You worked out the free trials. And you're on your way hopefully to your first sales. Now early on, you can think of sales as just like any other thing at a startup. Your goal is to - you don't have to do things at scale. In fact you can purposely do unscalable things to try and get early customers. That's the fun of it. But the other thing that I think is important to keep in mind is once you've done this enough - what you should start thinking about is what aspects of this are repeatable. And what aspects of this you know, are we going to scale further? And there's this - Christophe Chance is the really great blog post online about the five ways to build a hundred million dollar company. And he talks about he can a thousand customers buy a product that costs a hundred thousand dollars. Or he can have ten thousand customers buy a product that cost ten thousand dollars. Or he can have a hundred thousand customers by a product that cost a thousand dollars. And even though you don't need to know on day one which bucket you're going to fall into - most companies do fall into one of these buckets. And so you should start thinking about that as you're doing this if you want to be in the elephant category of a hundred thousand dollar product, that's great and you're going to have really high touch sales cycle and that's fine. You know that's Salesforce. That's work day. That's great. But if you think that you're going to be a rabbit and sales products for a thousand dollars a year to businesses and your sales process involves flying out to see them three time and eight demos and you know three months of redlining - then you probably have to rethink something.

And so I see a lot of startups most commonly in that who want to be the rabbits and sell for a low price product to businesses. Not thinking about how to do it in a scalable way and that's one area where you can get under water. Or it just forces you to increase your prices.

So this is how I think about different businesses. And it will be helpful for you when you get started and once you've done enough for the sales to say, "Ok. You know, where am I?" And the corollary to that is how do I have to price my product to be a viable business?"

So that is - those are some of the things I figured out along the way building, building sales now at a few different companies. And specifically on this very narrow stage of zero to one million. After you get to one million you'll find there's a million blog posts you know, about how to get from five million to fifty million. Or ten million to hundred million. But the zero to one step I wanted to really focus the presentation on that today because there's not as much written about it and it is something that I think is very ope to our founders. I figured this out just by doing it and I'm confident that if you're starting a company you can too. If for whatever reason you would like to do what I did and join a startup that's figured it out and hone your skills and hone your craft - we are hiring at Clever. So that's an option. But even if - and if that doesn't - if you do want start your own company and you have questions about sales I put my email address up here. And feel free to reach out at any time. I am happy to help.

So. Thank you.
Lecture 19: How to Pitch

http://startupclass.samaltman.com/courses/lec19/

Michael Seibel

Sam: Thank you very much! That was awesome! Alright so now we're going talk about a little more about how to raise money. Michael Seibel is first going to talk about how you have a pitch. And then Qasar will do investor role playing.

Qasar: Yeah so (19:19 inaudible)... mind blowingly you know new. It really is a basic blocking attack. And the one point we wanted to make before we get started is we actually don't spend a lot of time at YC focusing on this. The main reason is the best way you can make your pitch better is to improve your company. If you - if you have traction and your product is doing well - these conversations are like the investors want to see you succeed. And so if you remember anything its make your company better and the pitch will be easier.

We're going to spend the time in the three kind of sections. Before the meeting what Michael will kind of focus on will do kind of a role play what meetings actually look like and then we'll just wrap it up. We are going to do Q&A at the end. We'll kind of save five minutes. So if there is something we don't cover please write down your questions and we'll go through them.

Well - without further a due.

Michael: Alright. So, my name is Michael Seibel. I am a current YC partner. I started two companies. One was called Justin.tv. I ended up selling to Amazon. The other was called Social Cam - which sold to Otter desk. And what I really wanted to do was break down and demystifying the process of creating pitch. I think what happens to often when I see companies coming to talk to me is that they don't know simply explain what they do. And then ask for money. And that's basically what you have to do as a founder.

So, we're going to go over four things. The first is what your 30 second pitch is. This you need to be armed with constantly. This is basically how you talk about your company. This is magic. Whether you're talking to people who want give you money. Or don't want to give you money. You're talking to your parents - this is your go to.

The second is your two minute pitch. This is for people who are more interested. This is people who might want to raise money from or people who might want to get to work for. Or people actually kind of need to get a little bit deeper. Notice that's where I stop. A lot of people practice ten/ thirty minute pitches - hour pitches. I think that's all garbage. I think you can get everything you need done in two minutes. And one thing I like to tell founders is the more you talk - the more your opportunity to say something that people don't like. So just talk less and it will probably be better.

So then I want to tell you about when to fundraise. 'Cause I think a lot of companies get this a little bit wrong. And then quickly how to to set up investor meetings.

So - 30 second pitch. This is so simple. Its three sentences. You can take time. You can breathe when you do this. You don't have to get that much information out. The first is one sentence on what your company do does. Everyone I meet for the first time Screws this up. You have to be able to do it in a way that is simple and straightforward that requires no preformation on my part. You have to assume
I know nothing. Literally nothing about anything. This is how you make it super simple. So you know usually what we tell people is apply the Mom test. If in one sentence you cannot tell your mom what you do, then rework the sentence. There is a one sentence explanation that your mom or your dad is going to understand. So really, really start there. And it's ok if you use really basic language. It's ok if you're saying, "Hey we're Arabian B and we allow you to rent out the extra room in your house." That's simple! Right? You don't have to say, "We're Arabian B and we're a market place for space." I don't know what that is! That's going to require more time. So use simple language. Very, very important.

The second is in a multi-billion dollar market. It's pretty simple to do this. You know Arabian B might say, "How big is the hotel market? How big is the vacation rental market? How big is the online hotel booking market?" These are simple numbers to look up on Google. And it makes an investor understand, "Oh wait. If we're big - if we really blow this company up - it could be worth billions of dollars." Don't skip this up. Second sentence. How big is your market?

Third sentence - how much traction do you have? Ideally this sentence is saying something on the order of, "We launched in January and we're growing 30 percent month over month. We have this number of sales. This amount of revenue. This number of users." Very simple. If you can't speak to tracts in terms of prelaunch - you need to convince the investor that you're moving extremely quickly. So the team started working in January. By March we launched a Beta. By April we launched our product. Right? Convince the investor that you guys are moving fast. That this isn't some long slog. That you guys aren't thinking about this like a big corporation. You're thinking about it like a startup where you can move fast and make mistakes that's all you have to do in 30 seconds. Three sentences. From that basis you should be able to start a conversation about your company. From that basis I understand exactly what you do. You have no idea how valuable it is to be able to explain to someone what you do in 30 seconds. So really internalize that. Like if you take nothing else away - that's going to help you.

Ok. Two minute pitch. Now you got someone you actually have to convince of something. Maybe even someone you have to ask for money. So I like to add four additional components. And these also go by very quick. The first is unique insight. Now if you talk to VC's they'll say stuff like, "What's your secret sauce? What's your competitive advantage? What's unique insight?" It's all the same thing. When I think about unique insight, what I think about is here's your opportunity to tell me something that I don't know. Here's your opportunity to tell me something that the biggest players in the market you're trying to enter don't understand. Or don't do well. This is the AHA moment and you better have it down in two sentences. The AHA moment. So you got to crystalize all the reasons why you guys are going kill the competitors or the really intelligent thought that got this business started in two sentences. And I need to AHA. You can see whether it's happening when you're saying it. That's why I like two sentences so you get in and out fast. So if I look at you and I'm like, "Uh." Then it's ok. You nailed it. If I look at you and I'm like, "I already knew that." Then you didn't nail it. If I looked at you and I just don't understand what you're talking about you definitely didn't nail it. So practice that unique insight. In your two minute pitch that's all you're going to get - you're only going to get two sentences to get that out there. So it can't be complicated. And that's basically the theme of this whole thing right? It cannot be complicated.

Next - how do you make money? You know your business model. I see so many founders run away from this question because they think things like if I say advertising people are going to be like "Oh that's stupid." Just say it! Don't run away. If it's advertising - say advertising. Facebook's a massive advertising business. So is Google. If it's direct sales - it's direct sales. If it's you know a game and you're selling in app add ups - like that's fine. Just say it. Don't run away from the sentence. It only has to be one sentence long. Where founders get tricked on how you will make money is they say, "Well - we're going to run advertising. Maybe some virtual goods. We're going to figure out how to this. And maybe this. And maybe this." Well now you're saying nothing. Now you've told me you have no idea how
you monetize this. This was a check mark that I just wanted to write. And then I am going to monetize it - instead I am writing a bug question mark. So do the thing that everyone else your industry does to monetize 95 percent of the time - say it and move on. Like it's totally ok. No one's going to hold your feet to the fire and say three years later you didn't monetize this way. But it's much better to be clear and concise than it is to start spouting out every single way your company can make money.

Then next one is team. I think that this answer is actually really clear. I think you're trying to do two things. If your team has done something particularly impressive - you need to call that out. "We were the founders of PayPal." Probably want to say that. "We were the founders of Amazon." Probably want to say that. So if you guys have done something that is made investors money. You want to say that. If not, then please don't go on about the awards your team has one or the PhDs - I don't care. I don't care. What we want to hear is how many founders. Hopefully between two and four. We want to here is how many them are technical? How many engineers versus business people. Hopefully it's fifty/ fifty of more engineers. We want to hear is that how long have you guys known each other? We don't want to hear that you guys met a founders dating an even three days ago. Ideally you've known each other either personally or professional for at least six months. We want hear is that you're all working full time. It's really helpful. We're all committed to this business. And what we wanna hear is how you met. That's it. You can get in and out of that two sentences very easy. Your only way to build credentials is if you have accomplished something. And with an investor, typically if you accomplished something that's made someone some money. So don't try to over inflate yourself if you don't have that stat on your resume. Move on. The more you talk about a bad thing - the worse it looks.

So the last one is the big ask. When it comes to this and you have to figure out whether this is a conversation involves fundraising or not. What I tell people is like this is the time where you kind have to know what you're talking about. This is a time where you have to know are you raising on convertible note. Are you raising on a safe. You have to know what the cap of that safe is. You have to know how much money you're raising. You have to know what the minimum check size is. These are things where if you don't know these things, investors going be like, "These guys aren't serious. Or they haven't done their homework." So where's the rest of this whole thing you shouldn't use any jargon. And this part you shouldn't just be like "Oh we're just raising some money." Like now is time to actually use a little bit of that jargon. If you don't know that jargon - Google search it. Like it's real simple. You'll guys learn it fast.

So that's it by the way. That's two - that's all your pitch. Done. Like game over. Now you let them talk. So when to fundraise. I think this is so important. You got this little growth graph here. Investors like to invest based on traction. And so literally is always better to raise money when you got more traction than less. Often times though, you guys will be in a situation where you're just starting. Or maybe you just launched. So what you need to do is you need to think about how you flip the equation. You entire mindset should be typically you are the ones investors for money and therefore they strong and you are weak. How do create a scenario where you are strong and they are weak? Right? That's where you want to be fundraising. S first how do you know that you're strong. If investors are asking you to give you money - you're strong. That might be a good time to start fundraising. If investors aren't asking about giving you money - are you talking to people about your start up? Or are your running super stealth. If you're talking to people about your start up and you're getting the word out - either that's through the press or just through talking to your friends or people you know doing startups. That's a good way to kind of start feeding that. The second this is, have you created a plan so that you can launch and grow without needing to raise a bunch of money 95 percent of the startup that I meet can get a product to market with a very, very little bit of money. So never put the investor in the ultimate position of power. We can't do anything until you give us money. You always want to flip it around. You always want it to be, "This thing's moving. We all left our jobs. We're all working full time and it's moving... If you want to jump on great! If not there are a lot of angel investors." That's the attitude you want to have. That's
the confidence you want to have. If you need money early - always plan for needing less money. And always be able to show that you’ve got a fully committed team that’s working fast. That’s going to be how you gain an advantage when you can’t show traction. If you can show that investor that you haven’t launched yet but you’ve done eight months of work in one month or two months, then you’ve got a great team that have all quit their jobs and they’re totally committed - you get some of that advantage back. But you don't get all of the advantage unless you launched and growing. So something to keep in mind.

Finally how to set up investor meetings. This is really, really simple but I’m surprised at how many companies don't get this right. The first is you want a warm introduction from another entrepreneur preferably. Or a previous investor of yours. That’s where you want to start. If someone who's past on your company as an investor offers you to make introductions that's kryptonite. Don't touch that. So first warm introduction. Very simple. You don't want to cold call these people. You don't want to bum rush these people. The person - the credibility of the person who is introducing you to an investor is big part on whether the investor will take that meeting.

Second, think in parallel. So many people that I meet will run the fundraising the super slow process. We met with one guy this we. We're going to schedule a meeting with another guy next week. Another guy three weeks from now. When you're fundraising you're on. It's a sprint. It's not a marathon. So you want to schedule all of your meetings during the same week. It's extremely hard to do but here's one trick that I love - tell when you're emailing investors you getting those warm intros the investors email you back you say, "Hey we would love to set up a meeting but we're building like crazy for the next two weeks. So can we set it in that third week?" Right? So then you've emailed everyone that. Right? So everyone schedules that meeting three weeks out. It's better for them because their calendars open. It's better for you because you've got all you meetings in one week. And also what did you do? You hinted, "Hey. I am not desperate for the money. We're building. Like I can meet you in three weeks but we're building. We're busy." Like it's signally all of the right things. So, that's the best way to kind of go about how you're gonna do that. The last thing is one team member should be investing in fundraising full time. It shouldn't be something that takes over the whole company. Because it's very, very distracting.
Lecture 19: Investor Meeting Roleplaying

http://startupclass.samaltman.com/courses/lec19/

Qasar Younis, Dalton Caldwell

So with that - let's kick it off to the next part of this. Who am I handing it to?

Big Dalton (33:53 inaudible)

Dalton: Hi. So my name is Dalton Caldwell. One of the (34:01 inaudible) I'm one of the partners at YC and one of the things that we're going to do today real quick is a mock pitch. And first of all I know this is a bit contrived. This is - in this format of like a college class, we're going to do our best to have fun and kind of demonstrate what it's like. And I realize there's a million reasons why this - why you can say, "Of this isn't realistic of what pitches really like." But again there's a lot that we can show you.

Just in terms of my background - over my career I've raised 85 million over several companies so I've sat in a lot of investors meetings. So I'm going to be pulling as many things as I can. So again, we're just going to try to show you something to talk to and use it as a learning session. You already did your intro earlier Qasar right?


Dalton: Cool. So we're going to do two pitches and we're going to go through them pretty fast. As Michael said these tend to go fast. So let's go dive into the first one.

Ok. So Qasar I understand you, you're coming to pitch me today. What can you tell about what you do?

Qasar: So we're building a communication platform that will allow you know businesses and consumers to collaborate on one single platform rather than their kind of fracture state that they're in right now.

Dalton: I don't follow.

Qasar: So like think about what WhatsApp or Snapchat. That's for consumers. We want to do that for businesses. And so with that - I have to do this with a straight face. What that means is we want to enable consumers to talk to businesses. That's really the goal of our business or what our startup is.

Dalton: I still don't - so who uses this product? What does the product do?

Qasar: So I mean it's for consumers and businesses. A messaging product. Allows consumers to send

Dalton: Why would, why would a consumer want to use your product?

Qasar: Because they want to message a business.

Dalton: Ok. What can you tell me about the market - what the opportunity? What's the size of this company?
Qasar: Messaging companies are really big obviously. WhatsApp sold for like 19 billion dollars. And Snapchat is really growing very quickly as well. So we think the opportunity is very big.

Dalton: Ok so - Can you tell me a little bit about your traction, your numbers. Like have you given this to people yet?

Qasar: Yea. I mean we don't want to kind open the kimono and kind of go into all of the details here. I kind of had a high level hour live - we definitely have thousands of users in the Bay area. Hundreds of businesses you know of kind of....

Dalton: Can you tell me who some of those businesses are?

Qasar: There's ones that you've been to. We don't really want to get too much into the details because we're still early - we don't want you know - We're trying to stay stealth.

Dalton: Ok well, can you tell me about what you've learned so far. What insights that you've had from the customers...

Qasar: Yeah the consumers are sending messages to these businesses. And we think that's great. So and these businesses are responding to the messages and we think that's - I don't think that's obvious that would happen.

Dalton: So can you tell me about what your business model is and how...

Qasar: Yeah so we, we charge businesses like a monthly rate. We haven't precisely figured out what that is. We - right now we're free for the few hundred companies we're in right now. But we're looking to probably do a monthly...

Dalton: How much do you think a business would be willing to pay?

Qasar: We thing certainly ten to fifteen thousand dollars a month...

Dalton: Ok. So anyway can you tell me a little bit about your team and who you have working on this.

Qasar: Yeah we have five founders. Technically I am the only one who's full time. Right now. We're raising money. So we can get you know the rest of the team on board. Yeah

Dalton: Can any of the founders program or...

Qasar: Yeah. I mean we have - one of them has a Bio PhD but he's really picked up coding. The - I am a python developer. I did - I learned python the hard way.

Dalton: Look at the time. Well it's been really great meeting you.

Qasar: Yeah

Dalton: Please keep me in the loop. This sounds fantastic.

Qasar: I will send you an update
Dalton: Just keep me in the loop as this progresses.

Qasar: Yeah. I'll send you an update. Great - That was awful.

Dalton: Ok. So let's just go through

Sam: That's disturbing. You guys.

Dalton: So that was obviously not strong. So let's just talk about some of the mistakes. So first of all - you need to make sure the person you're talking to know what you do.

Qasar: This seems really simple but it's not

Dalton: Seems really simple but it's not. So many times people get flustered. They get nervous and they start talking really fast. And there's no way you're ever going to convince anyone of anything if they don't know even what your app actually is. You have to know your numbers obviously. If you're very vague or evasive - like don't even have a meeting. If you don't feel comfortable telling an investor what your numbers are - don't even meet with them. It means you're not ready yet. Right? For market size try to give some plausible bottom up analysis and don't just name drop big companies that aren't even really related to what you're doing. People tend to do that a lot. Try to have insights. Try to convince me that there something that I don't already know about the market that I learned talking to you whether than just - what everyone knows about what the market is I learned during that particular pitch. Also the team - Why are you working on this. Why are you suited for it? Is a good thing to do? And finally, like he didn't drive the conversation anywhere. Like obviously that went poorly and he just let the conversation just like flail around until I cut the meeting because ran out of time as fast as I could.

So anyway, that was not a good pitch. So let's try that again.

Qasar: Ok. Let's do this.

Dalton: Alright Qasar well I understand you have you have company and can you just tell me a little bit about what you guys to.

Qasar: Yes, so we're a messaging a product. We allow a tons that's kind of vague so what we allow you to do is essentially message a location. So when you walk into a Crate and Barrel, you can see the Crate and Barrel manager a message like "Hey. There's puke in the hallway." Or if you in the airport "I am trying to find this specific gate 'cause I am not at this airport. Where is the terminal for Virgin?" Or if you're at Target, "What aisle is the shampoo in?"

Dalton: And so is this a mobile app? Or what are you

Qasar: Yeah so on the consumer side we have the IOS and android app but really getting consumers to download apps is obviously very difficult.

Dalton: I don't usually just download apps to send a message to Crate and Barrel.

Qasar: I most businesses we have a call to action which says text the owner directly. We tested actually a bunch of copy that works the best in small print. In small print we have the messages are anonymous.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
They also lower the barrier to entry. I think that most counter intuitive then we've learned in the kind of launch that we've had - in three hundred fifty locations in Bay. We've been doing this for about three months. We're about 11 percent weekly growth rate in terms of requiring businesses but most counterintuitive thing that we learned - Because we weren't actually sure is - Will people send messages while they walk it work...

**Dalton:** Do people send messages

**Qasar:** and they do.

**Dalton:** Like what's the number one type of message that people send?

**Qasar:** So originally we started the product thinking this is going to be like in location feedback. That was the premise. In location feedback. What we found is more than half the messages are actually not about feedback at all. They ask things like, "We were in this location in San Jose - this khaabob stand - Father and Son and we say messages that went through the satellite like are you hiring? And that's like very strange because you would think like why wouldn't you just ask the owner? But we realized that we know this is the owner and the person who's walking in doesn't and so they do prefer to actually just text the owner because I think that's an easier reading.

**Dalton:** Ok so it's like a suggestion box. It's like a way to just like message a business

**Qasar:** Initially that's what we thought what it was. But what we actually discovered was vast majority of - I shouldn't say vast majority. Over half the messages are just things like, "When do you open? When do you close? 'Cause that's not on Google. Do you - are you catering? Do you have any reservations available tonight?" etc.

**Dalton:** Ok look - in terms of your traction is sounds like you said some businesses. Like tell me about what you guys have right now.

**Qasar:** So we have three hundred and fifty businesses - all from San Jose to San Francisco. We sold them ourselves as three founders. We're all technical but we actually did all the sales because we learned a lot about how these businesses work. We actually come from a retail background. We originally built this product for large enterprise players like Starbucks and Walmart but we recognized at closing those contracts and our limited amount of runway wouldn't really be possible. So we wanted to get the product in the hands of users so we did S&B's. And that's when discovered, hey this like messaging product...

**Dalton:** Ok that sounds interesting. It sounds like you have customers. How can this be big though? Like ok - maybe you can get whole thousands of words....

**Qasar:** So in terms of like numbers - we see one and half messages on average per location per day. That might not sound a lot but for a business that's getting thirty messages - you take like a Yelp review or a Google review in a life time of business they might get five or seven. So they're getting a huge volume of messages relative to what they tend to experience and they're private so they are not public. So in terms of how do we actually make money, it's not - you know frankly speaking we don't have a very clear answer there. The two pats are the S&B side or the LC side the large customer side. Large customers we know from a retail experience just regular feedback tools are are three to four million per per year. So like a Sears - where we came from. S&B's we've tested are willing to pay 50 dollars a month. So I, you know certainly I think this is - can be a large business but there's clear ways to make money but...
Dalton: I can see that. Just a couple things. Like, can you tell me about distribution strategy and also just a little bit about the team

Qasar: Yeah, so distribution - so the thing that we learned in selling through these S&B's is really freaking hard. The formula LTV minus CPA - Life time value minus Cost Proposition A in S&B is never going to work out. So we have two solutions - one is to go up market like we originally planned to Starbucks or Walmart’s. Or two is actually essentially pair with consumer facing companies Yelp, Google, Facebook...

Dalton: Have you been talking with them. Are they going to actually do it?

Qasar: Yeah - so we've talked to Google and Facebook. We're meeting with the Apple. We're basically want to introduce every time you search for a business there should be a message button. We want to get consumers in the habit of knowing they can send essentially a text message to any business. That can help us get broad distribution. Our real vision is to become kind of that infrastructure - that messaging infrastructure between consumers and businesses. If that doesn't work - Let's say Google, Facebook and Yelp don't want to give up that valuable property - it's really an add unit. We do just want to sell this an s feedback tool to large players.

Dalton: Alright. Can you tell me a little about the team - we're running low on time.

Qasar: There's three of us. All technical. Mike and I did a company before. Sonny was an ex school engineer. We come from retail. So our first start up was a failure. So I don't know if that's good or bad. We've worked together - we're all technical. We all built everything ourselves. And we sold everything ourselves.

Dalton: Ok.

Qasar: So we already had a couple of conversations with your firm. We're raising five hundred thousand on an 8.5 million convertible note. Of that five hundred two hundred -fifty is committed by Mike Maples, Eli Gill and Aden Sinket. And Mike with Floodgate is willing to fill the round. We think you're - you particularly - you and your firm can bring a lot to the team with your retail experience. Is this something that's interesting to you?

Dalton: Yeah - you know I think this is really interesting. I mean I would need to talk to a couple of more folks on my side but I do think that this - this could be pretty big.

Qasar: Yeah since we've had a couple of conversations before and we're certainly willing to meet again. We are closing a round this Friday and so certainly take time and let you other partners know. I will be available between now and Friday. I'll give you another call before Friday before we close the round. But we've love to actually see you - see you in the run.

Dalton: Ok. Well it sounds good. I got to go but thanks for that

Qasar: Great. Thanks.

Dalton: So in terms of that one you know - some key points here is try to actually tell a narrative that makes sense to people. You noticed there was narratives there talking about people - how they really use it. We were able to like tie it down to the real world. Which is good. He was able to demonstrate
insights and actually tell me something I didn’t already know about the market. Like there were some tidbits. It was more of a collaborative meeting where it felt more like a conversation than just like I was interviewing about something in my opinion. He actually asked for money. You saw I could have easily been just like, "Ok. Got to go." But he did talk about fundraising as Michael mentioned. And he was able to provide all he context and all the the questions to actually have a serious conversation with him. If he was KG about it or shy about and clear on the numbers there’s a very good chance I probably would have just ended the conversation due to time pressure.

**Qasar:** Yeah. It’s interesting we sit on this side a lot. You really - you can tell when people are very passionate and know their business very, very well. And that’s what you have to become.

Ok so closing thoughts here before we - what you want to do after the meeting. Before we get into Q&A. We’re running a little short on time.

After the meeting the first just like Tyler said in the sales things follow up. This is important. Anything other than a check or wired funds is a no. So they we got to keep talking to partners - I assume that’s a no. And so you do want to put some pressure. The way you can do that is get deal heat. A deal heat is just a term that means there’s a demand to be in your round. This is the easiest way and important way to drive a price, etc. Do diligence on investors, So let’s say you have that five hundred thousand to raise for your seed round on the 8.5 million like we used as an example, Do diligence on the investors - If you do find - I do the diligence on Dalton and I found that hey he’s actually not great investor, I can get Millan or Mike Maples or whoever to actually fill the rest of the round. It’s uprising to us how money entrepreneurs don’t do this. You would - it’s like you would actually spend a lot of time hiring somebody - you’re selling a part of your company to somebody you should know who you’re selling it to to make sure they’re the type of people you think they are. And then last - know when to stop. So some founders get so good a fundraising they just want to it all the time because it’s much easier to do than actually building the company.

**Dalton:** And you think - fundraising does not equal success. Just because you fundraise does that you succeeded and nobody realizes that. And I’m - we say this - and we’ll say this now but I am sure that everyone still equate fundraising with success and read about someone’s fundraising and assume that means we’re successful

**Qasar:** My guess - my intuition is why this is a case because a lot of smart people their whole life they like applied to good schools and applied to good jobs and they think fundraising is just another application that they can just kind of like check off. And building a company is much more ambiguous but - Anyways - that’s the session. I don’t know if we have time for - Oh I just underlined build you company - fundraising is not the goal.

**Sam:** Can you guys just stick around for a few minutes after to answer questions?

Thank you guys very much that was great!
Lecture 19: Five ways to build a $100 million business

https://medium.com/@chrija/five-ways-to-build-a-100-million-business-82ac6ea8ff9d

Christoph Janz

(This article first appeared on my blog. But I've heard that Medium is cool, and so I want to try it by re-publishing a popular post of mine.)

Some time ago my friend (and co-investor in Clio, Jobber and Unbounce) Boris Wertz wrote a great blog post about “the only 2 ways to build a $100 million business”. I’d like to expand on the topic and suggest that there are five ways to build a $100 million Internet company. This doesn’t mean that I disagree with Boris’ article. I think our views are pretty similar, and for the most part “my” five ways are just a slightly different and more granular look at Boris’ two ways. The way I look at it can be nicely illustrated in this way:

The y-axis shows the average revenue per account (ARPA) per year. In the x-axis you can see how many customers you need, for a given ARPA, to get to $100 million in annual revenues. Both axes use a logarithmic scale.

To build a Web company with $100 million in annual revenues*, you essentially need:

1. 1,000 enterprise customers paying you $100k+ per year each; or
2. 10,000 medium-sized companies paying you $10k+ per year each; or
3. 100,000 small businesses paying you $1k+ per year each; or
4. 1 million consumers or “prosumers” paying you $100+ per year each (or, in the case of eCommerce businesses, 1M customers generating $100+ in contribution margin* per year each); or
5. 10 million active consumers who you monetize at $10+ per year each by selling ads.

Salespeople sometimes refer to “elephants”, “deers” and “rabbits” when they talk about the first three categories of customers. To extend the metaphor to the 4th and 5th type of customer, let’s call them “mice” and “flies”. So how can you hunt 1,000 elephants, 10,000 deers, 100,000 rabbits, 1,000,000 mice or 10,000,000 flies? Let’s take a look at it in reverse order.

Hunting flies

In order to get to 10 million active users you need roughly 100 million people who download your app or use your website. This is of course a gross simplification, and the precise number depends on various factors like your conversion rate, how active your users are, churn, etc. But it doesn’t change the take-away: To get to $100 million in ad revenues, you need dozens of millions of users. I know of only two ways to achieve that (plus one mega-outlier which breaks all rules, Google). The first one is to have a product that is inherently social and has a high viral coefficient (Instagram, Snapchat, WhatsApp). The second one is a ton of UGC (user-generated content), which leads to large amounts of SEO traffic and some level of virality. Good examples of this second option include Evernote and MailChimp. If you’re an eCommerce business you might be able to acquire one million customers using paid marketing, but it requires huge amounts of funding.

Hunting mice

To acquire one million consumers or prosumers who pay you roughly $100 per year, you need to get at least 10–20 million people to try your application. This is – again – a gross simplification, but I believe it’s order-of-magnitude correct. To get to 10–20 million users you almost certainly need some level of virality, too – maybe not Snapchat-like virality, but some social sharing or “powered by”-virality. Great examples of this category include Evernote and MailChimp. If you’re an eCommerce business you might be able to acquire one million customers using paid marketing, but it requires huge amounts of funding.

Hunting rabbits

Most SaaS companies that target small businesses charge something around $50–100 per month, so their ARPA per year is around $1k. To acquire 100,000 of these businesses you need something in the order of 0.5–2 million trial signups, depending on your conversion rate. Let’s assume that your CLTV (customer lifetime value) is $2,700 (assuming an average customer lifetime of three years and a gross margin of 90%) and that you want your CLTV to be 4x your CACs (customer acquisition costs). In that case you can spend $675 to acquire a customer. If your signup-to-paying conversion rate is 10% that means you can spend $67.50 per signup (assuming a no-touch sales model where your CACs can go entirely into lead generation).

So how can you get one million signups for less than $70 each? Most SaaS products aren’t inherently viral, there usually isn’t enough inventory to make paid advertising work...
at scale, and cold calling usually doesn’t work at this ARPA level. There’s no silver bullet, but the closest thing to a silver bullet is inbound marketing – besides having a fantastic product with a very high NPS (net promoter score) and being obsessively focused on funnel optimization. I’ve written about this in more detail in my “DOs for SaaS startups” series: Create an awesome product, Make your website your best marketing person, Fill the funnel, Build a repeatable sales process. Another option is an OEM strategy (i.e. getting your product distributed by large partners), which can work but comes with its own challenges. Interestingly, hunting rabbits looks much less straightforward than hunting flies or hunting elephants. Why we have a strong focus on rabbit hunting SaaS companies nonetheless is something for another post.

**Hunting deers**

If you’re a deer hunter and want to acquire 10,000 customers paying you $10k per year each, most of the rabbit hunting tactics still apply. An ARPA of $10k per year usually isn’t enough to make traditional enterprise field sales work, and you likely still have to get 100,000 or more leads. The main difference is that when you’re hunting deers you can use an inside sales force to close leads, potentially also to generate leads. It also means that you can pay VARs and channel partners an attractive commission, although I’ve rarely seen this work in SaaS. SaaS companies sometimes start as rabbit hunters and expand into deer hunting over time. This can work very well and we’re very excited about these types of businesses, but to successfully execute this strategy, SaaS founders with a product/tech/marketing DNA usually have to bring in an experienced VP of Sales who has built an inside sales organization before.

**Hunting elephants**

Like it or not, most of the biggest SaaS companies derive most of their revenues from selling expensive subscriptions to large enterprises. Workday, Veeva, SuccessFactors, Salesforce.com, you name it. Jason M. Lemkin, another friend and co-investor, once said (I’m quoting from memory) that if you have a good solution for a significant problem experienced by large enterprises, building a $100 million business is relatively straightforward. After all, you only need 1,000 customers, and the $100k you need from each of them is less than they spend on the salary of one executive. I think there’s a lot of truth in that.

The other part of the truth, though, is that it may take you several years and millions of dollars to find out if you really are solving a problem (a.k.a. product/market fit), and once you’re at that point, you still need tens of millions of dollars or more to finance the enterprise sales cycle. This does not at all mean that elephant hunting isn’t attractive. It just requires very different skills, which usually means a founder team with enterprise sales DNA.

That leaves me with the million dollar – sorry, one hundred million dollar – question: Which other ways to build a $100 million business are there that I’ve overlooked? Let me know!

* If you have $100 million in annual high-margin revenue, you will likely be able to exit for $500 million to $1 billion or more. That’s the kind of exit most venture capitalists are looking for, although we as a small fund can achieve a great fund performance with somewhat lower outcomes.

** For eCommerce companies, which naturally have a much lower contribution margin than purely digital businesses like SaaS and are therefore valued at much lower revenue multiples, it makes more sense to target $100M in contribution margin.
20. Closing thoughts
Good afternoon. Welcome to the last class of How to Start a Start Up. This is a little bit different than other classes, which have been about things that you should think about at the beginning of a startup. Today we're going to talk about things that you don't have to think about for awhile. Since I'm going to not going to get to talk to most of you again before you get to post-product market fit stage, I want to give you the list of things that you need to think about as your startup scales. The list of the things that founders usually fail to make the transition on.

These are the topics we're going to talk about. Again, these are not writing code or talking to users, which means with a few exceptions that I'll try to note, you can ignore them until after you have product market fit. For most companies, these things become important between months 12 and 24. Post product market fit. Write these down somewhere and look back them when you get there.

So the first area we're going to talk about is management. In the beginning of a company there is no management. And this actually works really well. Before 20 and 25 employees, most companies are structured with everyone reporting to founder. It's totally flat. And that's really good. And that's what you want. And at stage that the optimal way for - that's the optimal structure for productivity. But the thing that tricks people is that when - when lack of structure fails - it fails all at once. And so what works totally fine at 20 employees is from 0 to 20 employees is disastrous at 30. And so you want to be aware that this transition will happen. And you don't actually need to make the structure complicated. In fact you shouldn't. All you need is for every employee to know who their manager is and every - and there should be exactly one. And in every manager should know who direct reports are. You wanna ideally cluster people in teams that make sense of course but the most important thing is that there just clear reporting structure and that everyone knows what it is. And if you want to make changes to it - people understand how to make changes or to hire someone. Clarity and simplicity are the most important things here. But failing to do it is really bad. So because it works in the early days to have no structure at all and because it sort of feels cool to have no structure - many companies are like, "We're gonna try this crazy new management theory and have no structure." What you want to do is innovate on your product and your business model. Management structure is not where I would recommend trying to innovate. So don't make the mistake of having nothing. But don't make the other mistake of having something super complicated. A lot of people fall into this trap where they think it's like you know people feel cool if they're someone's manager and if they're just an employee they don't feel cool. So people come up with this convoluted circular matrices management structures where you report to this person for this thing and this person for that thing and this person for that thing. But actually this person reports to you for this thing and this person for that thing and this person for that thing. But actually this person reports to you for this thing. That's a mistake too. So don't try to innovate here.

This is the first instance of an important shift in companies or in the founder's job. Before product market fit, your only job that matters is the build a great product. Or your number job is the build a great product. As the company grows and at about this you know, 25 or so employees size your main job shifts from building a great product to building a great company. And it stays there for the rest of your time. And this is probably the biggest shift in being a founder that every happens.

There are four failure cases we see all the time as founders become managers. So I going to talk about the four most common ones. The first one "Being afraid to hire senior people." In the early days of a startup, hiring senior people is usually mistake. You just want people that get stuff done. And the
willingness to work hard and aptitude matters more than experience. As the company starts to scale, and at about this time when you have to put in place the basic management structure - it is actually valuable to have senior people on the team. You know executives that have built companies before. And almost all founders after the first time they hire a really great executive, and that executive takes over big pieces of the business and just makes them happen - the founder says, "Wow! I wish I had done that earlier!" But everybody makes this mistake and waits to long to do this. So don't be afraid to hire senior executives.

The second mistake is "Hero Mode". So I will use the example of saying someone that runs the customer `service team. Someone who runs the customer service team they want to lead by example. This starts from a good place. It's the extreme of leading by example. It's saying, "You know what? I want my team to work really hard rather than tell them to work hard I'm going to set an example. I'm going to work 18 hours a day. And I'm going to show people how to get a lot of tickets done. But then company starts growing. Also they have the normal discomfort of assigning a lot of work to other people. So the company starts growing and the ticket volume keeps going up. And now they're have to do like 19 hours a day. And then 20 hours a day. It's just obviously not working. But they won't stop and hire people because they're like, "If I stop even for one day we're going to get behind on tickets." The only way to get out of hero mode in this case is to say, "You know what? We're going to get behind on tickets for two or three weeks. Cause I am going to go off and I am going to hire three more support team members. And I've calculated based off our growth rate that this is going to last this long. And next time I'm not going to make the same mistake. I'll get ahead of it and hire again." But you actually have to make a trade off. You actually have to say, "You know what? I need to hire more people and we're going get behind on other stuff." That is the right answer. The wrong answer is to stay in hero mode until you burn out. Which is what most people do.

Third mistake - "Bad Delegation". Most founders have not managed people before and certainly have not managed managers. And so the way that- the bad way you delegate is you say, "Hey Employee we need to do this big thing. You go off and research it. Come back to me with all the data and the tradeoffs. I'll make a decision and tell it to you and then you go off and implement." That's how most founders delegate. And that does not make people feel good and it certainly doesn't scale. A subtle difference but really important is to say, "Hey - You're really smart. That's why I hired you. You go off. Here the things to think about. Here's what I think. But you make this decision. I totally trust you. And let me know what you decide. That's how delegation actually works. Because I think - because Steve Jobs was able to get away with the former, and make every decision himself and people just put up with it. And every founder thinks they're the next Steve Jobs. A lot of people try this. But I for 99.9 percent of people this second method here works a lot better.

And then the fourth area it's just a personal organization one. When you are working on product - you don't actually need to be that organized in terms of how you run the company and how you talk to people about what they're working on. But if you fail to get your own personal organization system right - where you can keep track in some way of what you need to do and what everybody else is doing and what you need to follow up with them on - that will come back to bite you. So developing this early as the company begins to scale is really important.

Two other things that we hear again and again from our founders they wish that they had done early - and that is simply writing down how you do things and why you things. These two things - the how and the why are really important. In the early days you just tell everyone. "Employee, when you're sitting around having lunch or dinner, you know this is how we think about building product. This is how we push to production. You know, this is how we handle customer supper." Whatever. As you get bigger you can't keep doing that. And if you don't do it - someone else is just going to say it. But if you write it down and put it up on a Wiki or whatever that every employee reads, you as the founder get to basically
write the law. And if you write this down it will become law in the company. And if you make everyone read this - as the company hires a hundred and then a thousand employees - people will read this and say, "Alright. That's how we do things." If you don't do it, 'all be like random oral transition of whatever the hiring manager or their best friend that they make it their first week in at the company tells them. So writing down how you do things and the why - The why is the cultural values. Brian Chest key talked about this really well. Every founder I know wishes they written down both of these, the how and the why earlier to just establish it as the company grows. And then this becomes what happens. I think it's one of the highest leverage things you can do that people don't

Next area - "HR". HR is another thing that most people correctly ignore in the first phase of start up because again it's not writing code. It's not talking to users. But it's a huge mistake they continue to ignore it. And the reason I think most founders ignore it is they have in their mind this idea of like TV sitcom HR, you know. Awfulness. But it doesn't have to slow you down. Actually it speeds you up. Most founders will say out of one side of their mouth, "People are our most important asset." And the other side, "We don't want any HR." So what they mean is that we don't HR - we don't want like the bad kind TV HR. What good HR means is a few things. A clear structure. Which we already talked about you know a path for people about how they can evolve their careers. Most important, one of the most important things is "Performance Feedback." Again this happens organically early on. People know how they're doing. As the company gets to 25, 30, 45 people - that gets lost and it doesn't have to be complex. It can be super simple. But there should be a way that it happens and it should be frequent. You know people need hear pretty quickly how they're doing. And it should tie if tie if they are doing badly to where you get them out of the company. Or if they're doing well it should - there should be a clear path to how this ties to compensation. And that’s the next thing.

In the early days of a startup, people compensation is whatever they negotiate with founder and it’s all over the place. As you grow - it feels hopelessly corporate but it really is worth putting in place these "Compensation Bands". So a mid-level engineer is in this range. A senior engineer is this range. Here's how you move from this to this. And it keeps things really fair. Someday everyone will find out everyone else's comp. If it's all over the place, it will be complete meltdown disaster. If you put these bands in place early you will at least be fail. It will also save you a lot of crazy negotiation.

One thing that I think is really important when it comes to HR is equity. Most people get this right now for the early employees. They give a lot of equity. But I think you should continue to give a lot of equity all the way through. And this is one place that you investors will always give you bad advice. I think - not YC. But all other investors give bad advice here. Most do. You should be giving out a lot of equity to your employees. Now this dilutes everyone. Right? This dilutes you as the founder and the investors equally. For some reason founder usually understand this as good. Investors are very short-sighted and don’t want to dilute themselves so they'll like fight you over every equity grant. But, we've seen a lot of data at YC now and the most successful companies - and the ones where the investors do the best - end up given a lot of stock out to employees. Year after year... After year. So I tell founders like, "You should think about you know, for the next ten years you’re going to be given out 3- 5 percent of the company every year 'cause you just get bigger and bigger. So he individual grants gets smaller but in accurate it's a lot of stock. I think that this is really important to do if you value your people you should be doing this. Specifically you need to do this with refresher grants. And you should get a plan in place for this early. You know I think you never want an employee in place where they vested 3 out of their four years if stock and they start thinking about leaving. So you should ALWAYS stay in front of peoples vesting schedules. And you know how they plan early where you have refresher grants in place.

There are a lot of new structures that people have been using here. I personally like six year big grants - but six years of vesting. ‘Cause I think these companies take a while to build. There's pyramid vesting where you back weight someone's grant. So in year four they get a lot more of the vesting than year...
One. There's a concept - different names for it but something like continuous forward vesting where people's grants are automatically re-uped. Every year. At the same number of share. Whatever you decide, get an option management system in place at about this point. The normal way people do this is just someone keeps an Excel spreadsheet, I have seen mistakes that have cost employees or companies tens of millions of dollars because they didn't get this right. So there's really good option management systems or software and you should get those in place around this point.

The other sort of HR stuff to touch on - there are a bunch of rules that change around 50 employees. Common examples are that you have to start "Sexual Harassment Training and Diversity Training". There's a bunch of others as well. But just put a little pen in your mind that when you cross 50 employees there's a new set of HR rules that you have to comply with.

"Monitoring your team for burnout." Again, it's up to product market fit. It's just a sprint. Now it becomes marathon. At this point you actually don't want people to work a 100 hours a week forever. You want them to go on vacation. You want them to have new challenges and do new things. And if you let the whole company get burned out all at once - that is often a company ending thing.

This is also a good time to put in place a "Hiring process". Another thing that most founders regret is they don't hire - as soon as everything is working, I think you should hire a "Full time recruiter". If you do this early - that's bad 'cause you'll hire too fast. And that usually implodes. But most founders get behind the ball on this. There are a lot of sort of hiring process tips. For example, I think most companies - even til they get up to say 3 or 4 hundred employees - should announce every offer on some internal mailing list or something before they make it. Because like half the time you do that. Someone in the company will know something good or bad about that employee. And the companies that I know that have instituted this have been really happy. Also a good time to have a program in place to ramp up employees. So when someone starts you know what their first week looks like. How did they get spun up? How do they learn everything they need to learn? Are they going to have a buddy that's going to think through them? That's going help them think through kind of everything about the company.

Here's one that you do need to think about before the 12 to 24 month mark. Which is "Diversity on the team." The most common place this comes up honestly is people that hire you know all guys on their engineering team for the first 15 or 20 people. And at that point you get a culture in place that sort of takes on a life of its own. And most founders that I've spoken to that have made this mistake regret it. And they wish they had hired some diversity of perspective on the team earlier on. Engineering teams are not the only place where it comes up. But that's where you see it the most often. And if you get this right early, you'll be able to grow the team much more quickly over the long term.

The other thing to think about is what happens to your early employees. So a common situation that happens is the company past the early employees. You know the company - So like you hire a engineer who's a really great engineer but then as the engineering team grows, you need a VP of engineering. The early engineer wants to be the VP of engineering. You can't do that. And but you don't want the early employee to leave. They are an important part of the culture. They know a lot. People love them. And so I think you want you be very proactive about this. You know you want to like think about, "What's the path for my first 10 or 15 employees going to be as the company grows?" And then just talk to them about it. Very directly. Be up front, you know. Sit them down and say, "I want to sort of where you want to see your career go inside of this company."

Alright, so - "Company Productivity". This is something that you don't need to think in the early days because small teams are just naturally productive most of the time. But as you grow, it - the productivity I think goes down with the 'square of the number of employees if you don't make an effort.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
Because it's sort of one these like connections between nods. Every pair of people add communication overhead. And so if you don't start thinking about the systems that you're going to put in place when the company is 25-50 people to stay productive as you grow - things will grind into a halt faster than you can imagine.

The second word that matters most I think to keep the company productive as it grows is "Alignment". The reason companies become unproductive is people are either not on the same page and you know don't know what the same priorities are. Or they actively working against each other. Which is obviously worse. But if you can keep the entire aligned in the same direction, you have won well over half of the battle. And the way to start with this is just a very clear road map and goals. Everyone in the company should know what the road map for the next three or six months or a year - depending on where the company is in it life cycle look like.

You know a classic test that I love to give - is if I walk into a company getting - beginning to struggle with these scaling issues - I'll as the founders, "Like, if I walked around and pulled 10 random employees and asked them what the top three goals for the company are right now - would they all say the same thing?" And 100 percent of the time the founder says, "Yes. Of course they would." And then I'll go do it and 100 percent of the time no two employees even say the same three top three goals in order. And the founders can never believe it. Because they're like, "Well I announced it in all hands like three months what are goals were going to be. And how can they not remember?" But it's really important to keep reiterating the message about the road map and the goals. And almost no founder does this enough. And if you do it, you know the company will say, "You know, alright. These are our goals. We understand them and we're going to get them. “Self-organize around that. But if people don't know what the road map of the goals are, it won't happen.

We already talked about figuring out your values early but I want to reiterate that. 'Cause that'll also really help company make the right decision. If everyone knows what the framework to decide it - they'll make hopefully the same decisions if they're smart people.

You want to continue to be run by great products and not process for its own sake. This is a fine, fine line. Because you do need to put some process in place. But you never want to put process in place that rewards the process. The focus has to always be on great product. One easy way to do this that a lot of companies try is they just say, "We're gonna ship something every day." And if you do that - you know there's at least a continue focus on delivery. And then "Transparency and rhythm" in how you communicate are really important. Most founders wait way to long on these but having a management meeting every week of just the people that report directly to the founder and the CEO - critical. All hands meeting - not quite sure how often is optimal for those. At least once a month. Where you go through the results and the road map of the entire company. Really important. And then doing a plan every quarter of what we're going to get done over the next three months and how that fits into our goals for the year - also becomes really important.

I put "Offsite" up there. Because I don't think people do those nearly enough. A surprising number the successful companies we've been involved with do a lot of off sites. Where they take their best people for a weekend to a cabin in the woods or somewhere and just talk about what we want to be when we grow up. What are most important things to be doing? What are we not doing that we should be doing? But get people out of the office and out of the day to day. Everyone I know that does thinks they're well worth the time.

So the goal in all of this productivity planning is that you're trying to build a company that creates a lot of value over a long period of time. And the long period of time is what's important here. You can avoid all of this and just like with the authority of the founder - make sure the company ships a great

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
next version. But that won't work for version 10. It won't work for version 11. I really believe that the single hardest thing in business is building a company that does repeatable innovation and just has this ongoing culture of excellence as it grows. If you look at the examples of this - most companies fail here. Most companies do one great thing where the founder just pushes to get it done and then don't innovate that well on follow on products. And it really takes founders that think about how I am going to do this second thing - this really hard thing to get something like an apple that can turn out great products for 30 or 40 years. Or longer.

Alright - these are super tactical "Mechanics". This is definitely just too just put on a list and remember these things for later. Alright - in the early days. People basically ignore all accounting and they have like maybe if they're lucky a shoebox full of receipts. They certainly don't have anything that looks like financial report. This is is a good time to get it in place. You know when things are working say month 18 or whatever - you can do this with an outsource person. Just say, "You know what? We like to get our books in order. We want to start getting audits every year. We want to start a relationship with an accounting firm." Easy to do. Definitely worth it.

This is also a good time to collect your legal documents because it's easy to fix things now. So if you actually assign someone to go through and collect every agreement that the company has ever signed - Then when your landlord tries to screw you out of your lease and no one can find the lease - Which happens like half the time somehow. Someone will be able to find it. Also, you're almost certainly missing something. You know, some employee didn't sign their PIAA or whatever and you'll find it now - it's easy to fix now. Its gets really hard to fix like in the middle of your next round of financing. So again this is time to bring like a little of the order to chaos.

"FF Stock" is a special class of stock for founders that founders can sell in a later round without messing up the common stock valuation. It used to be that most people set this up right when they started the company. Founders fund sort of popularizes which is why it's called FF stock. But it became a really bad signal. Right that were obsessed with their own personal equity when the company had nothing - turned out to like fail most of the time. And so investors learned if founders pushed on this in the seed round it was a very, very bad sign. Most founders don't actually want to sell stock until the company is worth like a billion dollars or something like. So I think that you can actually safely set this up after things start working in the next financing round and then you can sell it two, three, four years down the road. But it's a good thing to remember by around the time you get to the B round.

"IP, Trademarks and Patents". Actually just IP and trademarks. So, you have twelve months after you announce something if you want to patent it. And if you miss that window, it's very hard to do. So eleven months after launch or first publically talk about what you're doing - is a good time to file provisional patents. We recommend people just file provisional patents. All that does is like hold your place in line at the patent office. And it gives you another year to decide if you want to patent something or not. It only costs about 1000 dollars. It takes way less effort than a full patent. And most of the time you'll know whether or not you'll need a fully patent a year later. But if you just do this one step, you'll at least have the option.

It's also a good time to file trademarks for the US and major international markets. Again, if you don't do this at this stage - most people end up regretting it. And while you're at it - a good time grab all the domains.

FP&A - Good time, also to think about someone to start doing FP&A. I think most companies don't end realizing where they knobs on their financial model are until far too late. And I think it turns out that if you have someone build a really great model of the business - And by really great, apparently Roelf Botha - who was the PayPal CFO and built their FP&A model - the top, like the top sheet of his
spreadsheet was 15 hundred lines just a level of the detail people build these to. But you can really optimize the business and understand it at a level that I think most people totally miss. Most people don't hire someone like this until their many hundreds of employees. I think it's worth hiring earlier.

Another thing that I think is worth hiring earlier that almost no one does is a full time fundraiser. Let's say you hire someone like really, really great and their full time job is to raise money for the company. You hire them after your B round. And you say, "You know what by the time we raise our C round, we want the valuation be double what it would have been otherwise." You almost certainly get better results than if you hire an investment banker or someone else if it's just someone internal with the company. And you end up paying way less money and take like literally half the dilution. So I think this is one these like slightly non obvious optimization that people just fail to make.

"Tax structuring". So this is another thing. Once things are working it would be worth you spending a little bit of time thinking about how you set up the tax structure for the company. I confess I don't know a lot about the details here 'cause I just find it personally really boring. But like somehow if you assign the IP to like some corporation in Ireland that licensing it back to the US Corporation. You end up paying like no tax. No corporate tax. But I know that you can only do that like relatively early on. And this ends up being a huge issue for companies that don't do it that compete with companies that do it you know once they're big public companies. So that's worth doing.

A lot of people through the class have talked about "Your own Psychology" as a founder. Here's what they haven't said. It gets worse. Not better. As the company grows you continue to osculate. The highs are better but the lows keep getting worse. And you really want to think about this early on and just be aware that this is going to happen. And try to, try to manage your own Psychology through the expanding swing that it's going through.

Another thing that happens as you begin to be successful as you go from being someone that most people rooted for - kind of the underdog. To someone that a lot of people hating on. And you know you see this first in internet commenters who will be like, "I can't believe this shitty company raised money. It fucking sucks. It's like awful. And it only bothers you a little bit. But then like journalist that you kind of care about it start writing this and it just goes on and on. This also will go on and on as you get more and more successful. And you just have to make peace with this early. But if you don't it will bother you all the way through.

This is also a good time to start thing about how long of a journey this is going to be. Very few founders think long term. Most founders think kind of a year in advance and they think, "You know what? In three years I am going to sale my company and either I am going to become a VC or sit on the beach or something." Because so few people make an actual long term commitment to what they're building - the ones that do have a huge advantage. They're in a a very rare flight class. And so this is a good time to like sit around with your co-founders and decide, "You know what - we're going to work on this for a very long time and we're gonna build a strategy that assumes that we're going to be doing this for the next ten years. Just thinking that way alone I think it's probably a very high leverage thing you can do for success.

Take vacation. Another common thing that we see is founders will run their business for three or four years without ever taking you know more of a day of vacation. And that works for a year or two years or something like that. It really leads to a nasty burn out if you don't do 1.

Loosing focus is another way that founders get off track. I actually this is a symptom of burnout. When you get really burned out on running business you want to do easier things or sort of more gratifying things. You want to go to conferences and have people tell you how great you are. You know what to do
all these things that are not actually building a business. And the most common post YC failure case for the companies we fund is that they are incredible focused during YC on their company - and then after, they start doing a lot of other things. You know they advise companies. They go to conferences. Whatever. Focus is what made you successful in the first place. There are a lot of reasons people lose focus. But fight against that really, really hard.

This is a special case of focus. As you start to do well - you will start to get a bunch of potential acquires sniffing around. And it's very gratifying. You're like, "Wow! I can be so rich. And I'll be so cool. And MNA negotiations feel really fun. This is one of the biggest killers of companies. Is that they entertain acquisition conversations. You distract yourself. You get demoralized if it doesn't happen. If an offer does come in - it's really low. You've already like mentally thought that you're done and so you take the offer. As a general rule don't start any acquisition conversation unless you're willing to sell for a pretty low number. Don't ever just check it hoping that you're going to have the one miracle high offer. If that's going to happen you'll know because they'll just make you a big offer before you can meet them. But this is big company killer.

And then - just a reminder to everybody - that things that kills startups at some level is the founders giving up. So sometimes you should quit but if you miss manage your own Psychology and you quit when you shouldn't that is what kills companies. I mean that is that is the sort of final cause of death for most of these startups. And so if you can manage your own Psychology in a way that you don't quit - don't get to a place where you need to quit or give up on the startup you'll be in a far far better place.

So "Marketing & PR" is something that we tell companies to ignore for a long time. Everyone thinks in the early days that the press is going to be what saves them. We tell them all the time it doesn't work that way. It's definitely true. You know press is not what's going to save your start up. But as you start to be successful - this is something that the founders themselves need to spend time on. So once your product is working - switch from not caring about this to caring about it a little bit. And the two most important things for the founder to do - the founders to do - figure out the key messaging your selves. Never outsource to your head of marketing or PR firm. You founders have to figure out what the message of the company is going to bet. And once you've set that it kind of sticks. Very hard to change this once the press decides how they're going to talk about.

The other thing is getting to know key journalists yourself. PR firms. Will always try to prevent you from doing this because they need to have a reason to to exist... And so they're like, "We're going to handle the relationship with the journalist. We'll just bring you in for interviews." No journalist wants to talk to a PR flag ever. They're so much happier to hear from just too just hear from the founder. I think the biggest PR hack you can do is to not hire a PR firm. Just pick three or four journalists that you develop really close relationships with that like you. That understand you - which you get. And then you contact them yourself they will cover every story you ever give them. And they'll actually pay attention and get to know you and care about the company. This is so much better than the normal strategy of having a PR firm blast 200 contacts that never read their emails with every piece of news. So this is something that I think is important to start doing.

This is also the time in a company when business development starts to matter. And so in the early days you can basically ignore anything that would be like doing deals. Except maybe fundraising and sales. You know this is a time when they're important. And everything or many things that you do like even fundraising. It falls under the category of doing deals.

So there are - here's my one minute crash course on this. There are five points that I think are important to understand here. We've talked about this a lot. Nothing will matter if you don't "Build a
great product". So assume that you've done this before you go try to get anyone to do anything with you.

"Developing a personal connection" with anyone you're trying to do any sort of big deal with is really important. For whatever reason, most founders fail to this. Or many founders fail to do this. But no one wants to feel like they're this transactional thing. That you're using them to get distribution for your product or to raise money or whatever. And so figuring out some way to actually care about this person and care about what you're doing with them. And not view them - you brave to in your own mind not just view them as this one off transaction. You have to actually care about them and what they're going to get out of this.

"Competitive dynamics" - So this is like basic principal of negotiation. Most founders learn this the first time in fundraising. But it actually matters for everything. The way you get deals done and the ways you get good terms is to have a competitive situation. You know if you don't do this dealt with party A you're going to do it with party B. It's not always an option but it usually is. And this like the single thing that makes deals happen and makes deals move.

Tyler talked about "Persistence" the last lecture. So I won't hit on that again too much other than to say you go beyond your comfort point here most of the time as a founder.``

And then the fifth point is that "You have to ask for what you want". This is another thing - I still have trouble with this and certainly most of the founders we do have - You know if you want something in a deal - just ask for it. Most of the time, you know, you won't get laughed out of the room and might get it. But you have to be - like at some point you actually have to say, "You know this what I'd like to do. Even if it feels aggressive or an over reach or whatever.

So I am going to close this part of the talk with an image. One of the Arabia B founders drew this on like a business card or something for another founder that starting a company and then I saw it once and took a picture of it. "Cause I thought it was such a good summary. And what he had tried to draw here was the YCombinator process as he remember it. And I love it 'cause it's like so simple and it looks so doable when it's written on a business card. But you're trying to find product market fit. You know you're trying to build a product and you're trying to close the gap between those two gears. The only way to do that is to go off and meet the people. You can do this without getting really, really close to your users. And then he drew this graph that sort of on a white board that at YC and gotten kind of sort one of the YC rites of passage. But that's the graph of how adoption goes for a new company.

So you launch on the press. You get a huge spike. It falls off to nothing. At some point at least one point things look like they're going to completely die and kind of dip below the X access. They recover a little bit, you have this long, long troth of sorrow before things work. In Arabian B's case, it was a thousand days before the graph started taking upward. You have these wiggles of false hope. And then finally, finally, finally, finally things start to grow. Three years later. So starting a startup ends being this very long process. It is - you know it can be very rewarding. It's definitely long but it is doable. That's what I love about that drawing.

So with that. I think I have about ten minutes left. I can questions on this or anything else in the course that we've covered. If anyone has some.

Yes
Audience member #1: You have that diversity being important but an earlier speaker said that diversity wasn't important and that you should just hire people that are very much like you and then trust them (inaudible37:00 )

Sam: So the question is how you square the device of diversity being important with earlier speakers saying that you want people that are very similar.

The difference what you want is you want diversity of backgrounds. But you don't want diversity of vision. Like where companies get in trouble is when they have people that think very differently about what the company should be doing or don't work well together. You don't want that. You do want hire people that you know and that you trust and that you can work with, but if everyone on the team comes from exactly the same background you do end up developing somewhat of a monoculture. Which often causes problems down the road. Not always. Some companies have been successful with that.

So what we tell people is hire people that you know and that you've worked with before. But try to hire people that complementary and aligned towards the same goal. Not people that are exactly the same. 'Cause you just get a better skill set.

Audience member #2: So what are some examples of ways to make up productivity (inaudible 38:07) on a personal level? (inaudible38:09) how do you do that on a personal level and also on an advance level.

Sam: How to keep track productivity systems. So, the one I use which I actually thinks works really well is I keep one piece of paper with my goals for sort of three to twelve month time frame. And I look at that every day. And then separately I keep one page for every day of my short term goals for that day. And so if I need to do something in like a week I just flip forward seven pages and I write down. And then I also keep a list of every person and what they're working on and what I need to tell them and what I need to talk to them about. What we talked about last time. So every time I sit down with someone I kind of the full state and a list of things for that person that works really well.

Audience member #3: So we talked a lot about the startups growing but most startups fail. Any advice for how to fail gracefully.

Sam: Yeah. Yeah. Great question. We should have covered that.

How to fail gracefully. So, most startups fail and Silicon Valley almost goes too far on how it loves failure. Failure still sucks. You should still try not to fail. And this whole like thing of like "Ahh failure is great!" I don't agree with but it will happen to most people most of the time and it's a very forgiving environment. As long as you are up front about it and ethical and don't let anyone get into bad situation. So if you're failing first of all you should tell your investors, And second of all you should not totally run out of money. What you don't want is blow up which a bunch of depths that the company owe and everyone showing up to work one day and the door being locked. You'll know when you're failing and you'll know the company - things just aren't going to work. And you should just tell you investors, "Like hey. Sorry. This isn't going to work." No one will be surprised. Like I expect to lose my - or I'm willing to lose my money on every investment I ever I make. I know that happens most of the time and the winners pay for it you know still with a factor of a hundred. And so it's ok. No one - people will be very understanding and supportive. But you want to tell people early. You don't want to surprise them. And you want - you don't want to like let your employees get shocked when they know they don't have job. You want shut the company down in a graceful way. Help them find jobs. Make sure you give

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
the two or four weeks of severance payment so they're not suffering a cash flow problem. All that stuff is pretty important.

**Audience member #4:** How many immigrant founders have you seen in YCombinator?

**Sam:** How many immigrant founders have we seen in YCombinator? In the last batch - I think it probably went up for this next batch. In our last batch 41 percent of the founders we founded we're born outside the US. From thirty different countries. So it's a pretty big percentage.

**Audience member #4:** I was just thinking - what do you think are the good places to start start up?

**Sam:** Apart from the Valley where do I think are other good places to start a startup. Well I still think the Valley is the best by a very significant margin. But I think it's finally maybe beginning to weaken a little bit because the costs have just gotten out of control. To be clear - if I was going to start a company I still wouldn't think about it. I still will pick Silicon Valley. And think if you look at the data of companies of over the last few years that is to wins by a lot. But Seattle, LA - Lots of places outside the US - I think all of these makes sense.

**Audience member #4:** Like places outside the US?

**Sam:** I hesitate to make recommendations because I haven't spent enough time in the cities to really have an intuitive feel. But like - you know as well as I do the common ones people talk about start up hubs. I just can't make a personal recommendation there.

**Audience member #5:** So when should the founders start to thinking about hire a professional CEO - a senior guy.

**Sam:** When should the founders think about hiring a professional CEO? Never. You - if you look at the most successful companies in tech they are run by their founders for a very long time. Sometimes forever. Sometimes they even hire professional CEO and realize that is not going to like build a great company and so Larry Page came to be CEO again. I think if you don't want to be the long term CEO of a company - you probably shouldn't start one. I am not totally sure about that. I think there are exceptions. But generally that the transition that I talked about today if you go from building a great product to building a great company being a founder for nine of the ten years is going to be about building that great company and if you're not excited about doing that - I think you should think hard about it.

**Audience member #6:** What are some of the most common and alarming warning signs you should be looking for when you're trying to make the shift from building great product to building a great company?

**Sam:** What are the most common mistakes to make when you're shifting to building a great company? I think I went through most of them here. I tried to put everything here that I see people mess up most of the time. Yes.

**Audience member #7:** Is there a way to get involved in the Yom community before getting accepted?

**Sam:** Is there a way to get involved with YC before getting funded? No and intestinally not. It actually - I say the one thing you can do is if you work at a YC company and then later apply - I think probably like - well not probably that definitely if you get a good recommendation from those founders will help.

http://startupclass.samaltman.com/ (gekregen van Inne ten Have inne@darwine.nl 06-42804208)
with YC. So you know, working at a YC company helps but there’s not much you can do to help. And that’s intention. Like there is no pre start up in a way that there is premed. You should just focus on whatever doing and then when you start a start up - there are `things like YC and others that are structured to help you. Most of the founders we fund we don't know at all before we do it. You know you really don't need to get to know us or get involved. We're all good that way.

**Audience member #8:** The statistic you saying now harder to get into YCombinator than getting into Harvard. So I am curious the criteria’s that you use to pick up startups. Does it change over time.

**Sam:** The question is what criteria to pick startups and has it gotten harder? Has it changed? You know we the two things that we need to see are good founders and a good idea. And without both of those we won't fund the company. But that hasn't changed. That is always been the case. The applicant pool to YC has grown quite a bit. But most of - a lot of the growth is people who shouldn't be starting start up anyone that are just doing it cause it is sort of the cool thing now. So you know if you're really passionate about an idea and the idea is good and you are smart and you get things done and your we executing - I still think you have a very reasonable shot at YC even though the headline number is bigger.

**Audience member #9:** There's a certain market that you’re really excited about that don’t necessarily know all about yet - is there a certain track you recommend or ways to

**Sam:** Sure - if there's a market that you’re excited about but don’t a lot about yet what should you do? Two schools of thought on this. One is to just jump right in. Learn it as you go. That's worked a lot of times. The other is go work at another company in the space or do something in the market for a year or two years. I lean slightly towards the second but as long as you are willing to really learn and really study and to get uncomfortably close to your users - either case would work. And I don't even thinks that it’s that much of a disadvantage. I think all things being equal go spend a couple of years learning about it in detail but I don't think you have to.

**Audience member #10:** I have a question related to YC - So I think YC did a fantastic job in promoting partnership in Silicon Valley. In fact (inaudible 46:06) I master a suite off of YC in the past few years. The (inaudible46:13 ) you guys 180 companies per year coming to the market it looks like its hard to follow each of the YC company any more. Do you think that this will create some - some people will walk away from YC because they can not follow large batch of companies and (inaudible46:30 ) the company had to be very polised and the firm had to be think of the world about idea (inaudible 46:37 )

**Sam:** Alright so I think the question is do I think investors are going to fund less YC companies as we grow. No. Definitely not. Like certainly the trend in this is the other way. We have more and more investors saying that half their portfolio is not YC companies and they look forward to the day where it's three quarters. No I don't think that's a problem at all. I think that so not on my top hundred problem list. The opposite of that maybe.

Alright. One more question.

**Audience member #11:** When should a group of founders raise a seed fund or first fund?

**Sam:** When should a group of founders raise some seed money? This is a great question. I think that - I think that in general it's nice to wait until you have the idea figured out and initial signs of promise before you raise money. Razing money puts some pressure on the company. Sometime pressure. And once you've raised money you can't be in this exploratory phase in definitely. You end up having to rush
and so like if you haven’t raised money and your idea is not working you can fall around and pivot until you really hit on the thing that’s working. But if you’ve raised money and your `idea doesn’t work - You’re in this oh shit moment. And you have to pivot and you pivot to whatever vaguely plausible idea is. And that’s bad. So I think if you can wait to raise any outside capital more than say like a hundred or 200 thousand dollars even necessary - but ideally not even that. Until things are working or at least pointed in the direction of working you’re way better off.

Alright thank you all very much! This was fun!