

# Halvar's guide to Entrepreneurship

version 0.1 - June 23rd 2026

Feedback, comments, corrections: Please send them to me!

Author: Thomas Dullien (thomas.dullien+entrepreneur@gmail.com)

## Introduction

I founded two companies — zynamics, which I ran from 2004 to March 2011 and which I sold to Google (GOOG), and optimyze, which I ran from 2019 to November 2021 and sold to Elastic (ESTC). The first company was bootstrapped, initially with no cofounder (an early employee received a big equity stake later), the second company had a cofounder and was venture-backed.

Neither of these exits were “successes” by Silicon Valley standards, but they were definitely impactful and possibly life-changing for founders and most employees.

I learnt a few things, and after talking to founders or would-be founders, I decided to share my lessons more broadly.

This document contains information, anecdotes, and insights. I hope that they will be useful for other founders. **It will also contain a number of falsehoods that I believe to be true.** Please make me aware of where that is the case.

Warning: I only have experience in software/SaaS B2B and know nothing outside of that. With only two companies as experience, nothing in this document should be taken uncritically as truth. I also have no experience with a company between size 15 and 3000 employees.

<b>Halvar's guide to Entrepreneurship</b>	<b>1</b>
Introduction	1
Reasons to become an entrepreneur	3
The choice of your target market	4
<b>Funding, Fundraising, Investors etc.</b>	<b>5</b>
Bootstrapping? VC? What other options?	5
Understanding risk-return payoffs	6
Venture fund napkin math	7
First-time founders and the risk-return tradeoff	7
Ways of addressing this	8
You are going the VC route - what does this mean?	8
Early Exits, and why they are harmful for the VC	8
Understanding the VC's job, and VC dynamics	8
The value of verbal commitments	9
Top-tier firms can wait	9
If you get funded, a competitor will get funded, too	10
The importance of “momentum” and managing the perception thereof	10

Investors are neither your friends nor your therapists	10
How to run a fundraiser	11
How NOT to run a fundraiser	12
Investor updates and "potential investor" updates	12
Picking the right VC	13
The VC claim of providing "value-add"	13
The "big gorilla" theory of picking VCs	13
Your relationship with the GP	14
"Why aren't you raising now?"	14
Early-stage valuation	14
<b>Product and customers</b>	<b>14</b>
Technology vs. product, starting from a problem vs. from a solution	15
Product ideation	15
Top-line growth, bottom-line-growth, and everything else	16
The user persona and the buyer persona	16
How does this product get (user/buyer) promoted?	16
A shining example: The FinOps stunt	17
Development partners	17
The importance of clickable prototypes	18
Design is not how it looks, Design is how it works	18
<b>Hiring</b>	<b>19</b>
Interviewing	19
Managing people	20
Conflicts in teams and the "capuchin monkey hierarchy"	20
Having to fire people	21
Recognizing expertise where you don't have expertise	21
<b>Marketing</b>	<b>21</b>
<b>Sales</b>	<b>22</b>
How does this sales stuff work?	22
Managing a funnel	22
When to hire a sales professional	22
<b>Running the business</b>	<b>23</b>
Your three balance sheets	23
Cofounder relationships: The inverse marriage	24
Make room for non-operational conversations	24
Splitting equity between cofounder	25
Relationship between founders and investors	26
Consider getting a coach	26
You and your employees	26
Equity and ownership with employees	27
<b>Delegating work and personal growth</b>	<b>27</b>
<b>The end, the exit, the denouement</b>	<b>27</b>

## Reasons to become an entrepreneur

If you think about becoming an entrepreneur, think about the “why”. Do not expect to find a simple answer, but doing introspection is a good idea. I never really aspired to become one; I stumbled into my first company by accident: I wrote a piece of software that people liked, and then needed a legal entity so I could sell it. I realized afterward (during my time at Google) that beyond 4-5 years, I have real trouble fitting into very large organizations run by someone else.

Your reasons will be idiosyncratic to you. I certainly did not understand my reasons very well until **after** my second company; I only embarked on more introspection at that point.

My reasons were:

1. Some say work happiness is achieved by the triad of Autonomy/Mastery/Purpose. This is true for me for technical work, but when working in an organization, I also need a lot of **agency** to be happy. This means having the ability to fix things that I perceive as broken without running a long political campaign. In many large organizations, fixing even relatively trivial brokenness gets hard unless you're extremely high up. Having my own company gives me a ton of agency. The hiring process seems broken? Great, let's do a new one. There's poor code hygiene? Great, let's build a company that writes great code.
2. I love working with great people on problems just on the verge of the possible. This can be hard in large organizations, as large orgs tend to (unintentionally) **avoid** having too many good people on one team (the value for the large org is maximized by spreading out great people to guide many teams - so most teams will only have 1-2 great members). It is often easier to assemble a great team in a small company.
3. Great people are sometimes oddly-shaped — perhaps they don't have the right academic credentials, only work three days a week, live in strange places, work strange hours, interview poorly, etc. - but they are great. Not being able to hire great people due to some bureaucratic obstacle grates on me, and I love the flexibility of just being able to hire them. Large organizations need standardized components, and small orgs can work with “nonstandard parts” that are often unusually great at multiple things.
4. In order to function well in a hierarchy, I need to respect my immediate manager, and the majority of people above me in the hierarchy need to appear to me to be both competent and of high personal integrity. This isn't always easy to achieve in large organizations. Concretely at Google, I found Sundar an extremely uninspiring CEO, and the trend in upper management was a steady decline in both competence and integrity.
5. I have a(n un)reasonably high opinion of my own value, and that perception does not always agree with job ladders, salaries, promotion processes, and company politics.
6. Alignment of technical, economic, and ideological interest: I really love to work on things that manage to align my technical interests, my economic interests, and my ideological interests. Ideally, I want to work on something that I perceive as being super interesting, something that is economically valuable, and something that has an impact in the broader world that I perceive as positive. This can be very difficult to achieve in a large company.

7. Alignment of customer, company, team, and personal interests: As a middle manager in a big company, I often found it difficult to align “what’s best for the customer”, “what’s best for the company”, “what’s best for the people reporting to me”, and “what are my own interests”. In a startup, if done right, these things are auto-aligned.
8. There’s a part of me that loves building a product that delights users, that makes eyes light up with a mixture of “wow this is easy” and “I didn’t know you could do that”. I have a child-like joy in demo’ing “look what we have built!”, and I take great personal satisfaction in seeing a customer happy. In my heart, I am a people pleaser, and a good product is a win-win — the customer happily pays for it because it helps them. Finding that direct feedback from customers (and building delightful products) is hard in big orgs.

Your reasons may be very different. Mine changed -- when I started my first company, #8 and #2 were paramount because I had no experience in large organizations. That’s normal. Either way, it is important to think about “why do you want to start a company”, and also “why do you want to start **this** company”. The process of running a company has stretches where it really isn’t fun - in those moments you need to know why you’re doing this (to yourself).

Weigh the alternatives carefully — “would I rather do this for the next 7+ years, or ...”. Starting a company and having second thoughts 18 months into the journey is not a great situation.<sup>1</sup>

In the end, the first thing you need to do is “figure out what you want” - imagine yourself in 2, 5, 7, 10 years, and think about where you want to be at each of these points. Figuring out what you want is crucial, because everything else follows from that - and first and foremost the choice of your target market.

## The choice of your target market

More than anything else, your target market will dictate a lot of things about your company, perhaps the majority of things.

You should think carefully about your target market before you start, and it should be a very conscious choice. My first company was built in an extreme niche market (because I never thought about choosing a market, and just built where I was standing), and that had many drawbacks. My second company was in a carefully-chosen market, and a lot of things became much easier.

Broadly speaking, target markets come in 4 forms:

1. **No target market.** The market is so small that it’ll amount to less than 1m a year if you saturated it. This isn’t a company, it’s a shop.
2. **A growing niche:** Any market between a few million and less than 50m, with a sensible growth rate. My first company was built into such a market (it amounted to perhaps 6m when we started, and grew to 30-50m today - 20 years later). Sometimes a niche can explode into an enormous market, so a “good” niche can

---

<sup>1</sup> The pros and cons here can be a good basis for your pitch deck if you need to raise funding.

become (4) - but be careful, because many hyped niche markets never became huge.

3. **A healthy specialized market:** A market somewhere between 50m and a few billion — big enough to build a large mid-sized company (the German “Mittelstand” thrives here — these are privately-held businesses serving some specialized part of an industry).
4. **A huge market:** A market size that in 6-7 years will have dozens or even hundreds of billions, where even getting a few percent of the market will yield annual revenue in excess of 200m. This market can be tiny today if it is clear that growth will be explosive.

The type of market you're entering will determine how you will (need to) finance the company.

One note: Your target market can be much bigger than your market for your initial product. It is very common to use a much narrower slice of a market as a wedge into a larger market.

## Funding, Fundraising, Investors etc.

This chapter is longer than it should be, largely because I made so many mistakes in this area - and as a result accumulated a very long list of “things I should've known before raising funds”.

### Bootstrapping? VC? What other options?

**Market type 1 (“almost no market”)** is a trap, and almost certainly to be avoided.<sup>2</sup>

**Market type 2 (“growing niche”)** is a good niche market to bootstrap into. The market is too small to attract aggressively-funded competitors, and competition in this market may be moderate. If ...

... you have special insights into the market

... can make customers happy

... are OK with spending many years building a commanding position in this market

... think you can hire and retain enough good-enough people on the meager prospects of that market (some markets are very interesting/fascinating which can compensate some of the financial aspects)

then go for it. The benefits of bootstrapping into a fascinating niche market are many, but you need to be aware of a few caveats:

- Exit options can be limited — you won't IPO, and any acquisition will be either a technology or a team acquisition. Even if you decide to never exit, you will eventually need a succession plan.

---

<sup>2</sup> Caveat: If you're happy to work alone or in a team of two your entire life serving that market, it might work for you.

- Bootstrapping means you'll have to work with constrained resources, and you'll compete with many other opportunities in the world that draw away talent.

- The big risk of a bootstrapped business is becoming a trap - demanding excessive energy from founders while generating just enough revenue just to stay afloat. "Too little to live, too much to die" is the failure mode here, which will risk eating a lot of your **remaining life time with little payoff**.

On the plus side: There are many such markets, and the outcome for the founder can employees can be pretty great - an interesting example is the history of IDA Pro, the industry standard disassembler: Starting into a market of perhaps 3-4m, the founder grew his product and with it the market to 20+m in revenue and ultimately sold most of it to a PE fund, while both keeping control of his destiny and a commanding near-monopoly position in the market. It was a 25+-year journey though.

**Market 3 ("healthy specialized")** is a tricky one: The market will have healthy competition, but the comparatively small market size might spook traditional venture investors. This is a realm for certain types of angel investors or some VCs that call themselves "capital efficient"; in essence investors that prefer a different risk-payoff tradeoff than traditional VCs (who are in it to chase power-law exceptional outcomes - see next chapter). Often, owners of existing larger mid-size companies are partial to investing in such endeavors. These companies will often remain private forever. Weirdly enough, a lot of high-end industrial suppliers, a lot of aerospace and defense companies, and a good chunk of B2B software firms live in this sector. They often stay private for decades, with exits often to larger conglomerates or to private equity funds.

**Market 4 ("huge")** is the typical market for a venture-backed startup. Taking even a small fraction of these markets is sufficient to generate enough revenue for an IPO, and the market size is going to attract many competitors with lots of funding. The entire experience will be supercharged with a lot of external capital, and the risk-payoff tradeoff is most risky here. For better or worse, you are also starting a company over which you will almost certainly lose control. The few cases where dual-class shares have allowed founders to retain control are extremely exceptional, so you need to be very aware of your own motivations to go this route. If your principal motivation is that you want to be in control of your company, this might not be the right path for you.

In some sense, market 4 will mean you need to build a rocket ship, load it up with lots of very explosive volatile fuel (in the form of external capital), and then make a hail-mary run for the stars. That does not mean you can YOLO anything — you will need to be as careful and deliberate as with the other markets - but your odds of success are lower (compensated by higher return in case of success).

## Understanding risk-return payoffs

As you walk along these market choices, the potential return of a successful venture **grows exponentially**, while the probability of you individually realizing that outcome **decreases exponentially**.

Let us assume for a second that our societies are reasonably efficient at allocating capital to different ventures. This then means that the expectation value of all investments in all markets are approximately the same (otherwise capital could be more efficiently allocated).

At the same time, there are many ways of realizing the same expectation value: A safe 100% certain 1m dollars has the same expectation value as a 10% chance of 10m dollars or a 1% chance of 100m dollars, or a 0.1% chance of a billion dollars. Investors / capital allocators get to invest across portfolios of many bets, so they can expect to actually realize the expectation value.

It is important to realize that as a founder, you are not diversified. Mathematically, you should not expect the expectation value, but rather a median outcome (perhaps adjusted to 80%th percentile if you think of yourself as that good). And the median outcome is close to zero.

## Venture fund napkin math

A grossly simplified idea of an early-stage venture fund is as follows:

Someone raises USD 100m to do a seed fund. They invest 20 times 5 million into various startups. They know ahead of time that 18 of their 20 investments will go nowhere. The fund runs for 8 years, and if you want to have a great career as a VC, you want to generate somewhere around 20% of internal-rate-of-return. At 20%, money doubles every 4 years. This means that after 8 years, your money needs to have doubled twice, e.g. the fund needs to return 400m.

This means that the two remaining investments need to return 200m each. If both companies exited at 1bn, it would still not be enough, because the initial stakes (conventionally around 20%) would have been diluted from further funding rounds.

Venture investors need a few outsized outcomes to truly make their math work. Venture investors are at the far end of the risk-return curve. They also have an incentive to play repeated rounds double-or-nothing, provided the odds of "nothing" are just a notch below 50%.

## First-time founders and the risk-return tradeoff

The above dynamics mean that first-time founders will rarely have the same risk-return preference as their VCs. For most first-time founders, a small number of millions make an enormous difference in their financial safety. This misalignment in risk appetites is at the heart of many founder-VC conflicts, and various ways exist to address these conflicts at later stages (secondaries at series B etc.) - more on this below. For now, it is important to understand: If you are a first-time founder, and you have decided to go into a category 4 market, you are going all-in on a very slim bet. Accept this now, wholeheartedly. It is best to be clear-eyed about all your decisions.

## Ways of addressing this

Many VCs are very smart, and not ignorant of the risk-return tension. As a result there are a variety of constructs to address it, particularly in later rounds. If the business is growing fast and looks like a winner, founders will almost certainly have the opportunity to take **some** amount of liquidity (through secondary transactions, e.g. selling some stock as part of a funding round), improving incentive alignment. This is not customary in early rounds, but come Series B or Series C, it becomes much less unusual.

## You are going the VC route - what does this mean?

Congratulations, you are now a rocket ship builder. See above.

## Early Exits, and why they are harmful for the VC

To some extent, the most harmful thing to happen for our fictitious VC is **not** any of the 20 bets going to zero, but rather one of the remaining potential winners selling too early or too cheaply. This is where many first-time founders get tripped up: They think that returning a small multiple of the initial investment should make every investor happy. This is not the case: Angel investors and VCs need big multiples of the initial investment to make their side of the deal work. And neglecting this is a good way of making everybody involved pretty unhappy. The investors didn't sign up to make a bit of money. They signed up to swing for the fences.

## Understanding the VC's job, and VC dynamics

Interacting with VCs can be confusing. "Everybody screws up their first time taking venture" is a statement I have heard, and I would agree that this is often what happens. The precise job incentives and social mores of the venture industry are very different from many other industries, and few first-time founders understand this properly.

Fundamentals to know:

- 1. The job of a VC is not to give you money, until it is clear you will get the money you need from somewhere, and then to convince you to take their money instead.**
- 2. VCs prioritize overall portfolio returns, not individual company success. With limited time and resources, they naturally focus on their most promising investments.**

These two dynamics lead to a pronounced "feast/famine" dynamic when it comes to fundraising: You'll either struggle for funding, or be spoilt for choice, and your status can flip quickly between the two, in both directions.

The rapid reversals in the power dynamics between founders and investors can be very jarring and strenuous, particularly if they come as a surprise. From my conversations with founders, I've seen the following situations where dramatic reversals of power dynamics can occur and surprise founders:

1. A company looks promising, and the founders are being wooed by multiple funds with term sheets. The founders decide on a term sheet to sign, often thinking "oh great we now have this powerful partner". Founders are then confused if their investors treat the process of converting a term sheet to the actual funding documents as a zero-sum game, and negotiate accordingly (read: unexpectedly aggressively / hard-nosed). Founders may think "why are they so aggressive, aren't we business partners and all benefit from cooperation?" - but the investors are doing their job by trying to maximize their expected return, which involves maximizing their leverage should things turn ugly.
2. A company has multiple highly interested potential investors, and some external event leads to one of them getting cold feet. This can easily trigger a stampede where all of them get cold feet.

Fundraising has significant herding effects. The herd may stampede toward you and away from you and turn around at short notice.

## The value of verbal commitments

VC investors fall onto a broad spectrum when it comes to the reliability of their word. Some investors claim that they will always stand by their word, others don't make that claim; but often claims are decorrelated from the actual behavior.

For a VC, it is usually beneficial to be able to say "no" in a way that the founder hears a "yes but not right now". So not only do some investors not mean everything they say, founders will also often hear a different message than the precise words the VC said.

For a founder, it is safest to treat everything any investor says as ephemeral and subject to change for any reason (and a reason may be that a second has elapsed). "When the facts change I change my mind" is a good habit for investors, which also means that **nothing is real until you have a signed paper in hand**. Paper is also less ambiguous - what is signed on paper is usually not just your brain hearing what you want to hear.

At the same time, VCs **will** try to hold you to every little thing you say, so be very careful about the commitments you make verbally, and be very clear in caveat'ing anything that may change.

**No founder should get emotionally attached to any verbal commitment from any investor until it is on paper and signed** - maintaining emotional distance is crucial here, otherwise you spend a lot of energy mopping up your own disappointment.

## Top-tier firms can wait

Top-tier VCs have the luxury of time. Usually, their brand is so valuable that founders will give them preference even if the top-tier VC comes in late or with a less appealing term sheet.

As a result, it is rational for top-tier VCs to let lesser VCs do a lot of the legwork, and use the criterion "can this company trivially get funding from other VCs?" to screen candidate investments.

There's a Robbie Williams song with the lyrics "I love you dearly but baby she's Madonna", describing a man breaking up with a woman because Madonna wants to date him. **For top-tier funds, it is a rational strategy to swoop into very late-stage fundraising rounds and convince the founders to ditch other investors**, without whom the top-tier investor would've never looked at the deal.

If you get funded, a competitor will get funded, too

The "herding" effect has another implication that founders may find jarring:

You might not get funding, but if you **do** get funding, you will leave a group of VCs that you have convinced of both your market **and** your idea that are now keen to invest in the same idea in the same market. **You can therefore assume that they will go out and fund a competitor** (who will have an easier time fundraising than you).

Do not take this personally. It is a logical and almost inevitable result of the incentives.

The importance of "momentum" and managing the perception thereof

Success for venture-backed companies often relies on the ability to raise large amounts of capital rapidly for follow-on rounds, should a product get traction and "have legs". The more a company looks like a rocket ship, the easier it is to raise funding (duh), and the easier it is to raise funding, the easier it is to succeed and to actually build a rocket ship.

This means that an important job of the CEO / founders is to **manage the perception of momentum**. You want to be (or at least appear to be) in a position of strength. One of the worst things you can do is to create the impression that you have difficulty obtaining funding. The ideal look is the founder that has to fend off armies of zombie-like investors trying to storm his office to try to force the founder to take their money.

Investors are neither your friends nor your therapists

The necessity of managing the perception of momentum also means that you need to treat your investors and potential investors in the same way you'd treat potential customers: Be forthright, honest, but at the same time consider carefully what psychological effect your information will have on someone that worries about losing money.

Investor herding behavior means that their de-facto behavior is a bit like interacting with an animal: If you have negative emotions, doubts, or anxiety, you can easily spook them. As a founder, you **will** have emotional ups and downs, and your investors are not the people to share your emotional state with **unintentionally**.

Being a founder can be very lonely. Find other founders to kvetch about the hardships to. It is tempting to talk to your investors (which may be ex-founders) for emotional support, but it's often the wrong move. Build a support system far away from investor gossip.

**This does not mean you shouldn't go to investors for help**. Just make sure you have carefully thought through your options, are sure that you're not just in a mood swing, and do so **intentionally**. Investors can be very helpful when the time and situation is right.

## How to run a fundraiser

We understand that investors don't want to invest unless other investors want to invest, and that maintaining a perception of momentum is paramount. The mechanics of a fundraiser result from that.

People talk about "running a structured process", without necessarily explaining what this means. I'll try to describe what I think are the ingredients for a successful fundraiser are:

You need to be raising from a position of strength <sup>3</sup>if at all possible. You want to induce FOMO. This means you have the things that are appropriate at your current stage - be it a prototype or other strong evidence at Seed stage, or real traction with some revenue at Series A, or a repeatable sales motion at Series B etc. - try to avoid raising from a position of weakness if you can at all avoid it. A fundraiser from a position of strength will be quick, meaning:

- a. You're preserving your own energy.
- b. You're preserving the perception of momentum.
- c. You'll get better terms.

A fundraiser from a position of weakness on the other hand will...

- d. Take **a lot more** out of you in terms of energy.
- e. Take longer, hence run the risk of damaging the perception of momentum, which can be deadly.
- f. You'll get worse terms.

The idea of a "structured process" is to have collected contact information from a good number of potential investors ahead of time. When it is fundraising time, you need to prepare your pitch deck carefully (more on this later), and then you need to perform practice runs.

One philosophy I have heard (although some experienced founders dispute the utility) is to arrange the potential VCs in ascending order of preference. You then contact the first third of potential investors that you care least about, and tell them: We are running a fundraiser, we intend to have the round raised by date X (perhaps 1-2 months into the future). Are you interested in hearing our pitch?

You then try to schedule as many pitches as possible in one week, while lining up the second third of investors for the second week etc.

After each pitch and follow-on discussion, you improve your pitch and perform slight modifications to your deck and business plans. This way, you will be practiced when you reach the last third of investors (that you actually care about a lot).

One criticism of this approach is that pitching to not-great investors will not prepare you for the questions great investors pose, so YMMV.

---

<sup>3</sup> Good ways of signaling strength: Cash-flow positive quarter before the raise, important customer signing, hitting milestones ahead of schedule.

Either way, the hope is that at some point in the 3-week pitching process, you will first get a verbal commitment or a term sheet for an investment from one of the investors. If there are investors that you'd prefer to have, you can then reach out to them, and tell them that you have an offer from X on terms Y, but you'd rather work with them. In the VC pecking order, the lower-tier investors are incentivized to give you a first term sheet, and with one term sheet in hand it becomes easier to reach the higher-tier ones.

In an ideal scenario, you end up with multiple competing term sheets with variance in the terms that are being offered, and you can negotiate the terms most suitable to your (and the businesses) needs and find the investor that offers the best overall package that maximizes the odds of success.

A related advantage of this approach is: The lower-tier investors often know they are lower-tier and they know the way to get the deal is to be very fast - so they are more likely to give you a term sheet that you can then use to get term sheets from other investors.

## How **NOT** to run a fundraise

**Avoid an unintentional, drawn-out fundraise.** This can happen if enthusiastic potential investors tell you "why don't you raise now?" when you are not ready - either your numbers are not good enough, or you're not prepared to put in the time and structure to run a disciplined process. Unless you're under the immediate threat of running out of money, it's almost always preferable to **strengthen your hand before a fundraise**.

If - instead of creating a FOMO situation for investors - your pitch deck gets circulated between a small number of investors for a long time, without anybody biting, you're in trouble.

For investors, the perception of momentum is very important, because momentum is often responsible for success. The company that appears to have the most momentum can attract the most funding (and the best talent) at the most favorable rates, so in Silicon Valley, the **perception of momentum often translates into actual momentum** and actual success.

It is crucial to avoid creating the perception that you have lost momentum. This means that if your pitch deck gets circulated in January to a few investors, and you end up not raising, and you try to raise in July, eyebrows will be raised: What is wrong with these people that they have such difficulty raising? Clearly they cannot be winners?

Keep your fundraises short, sweet, structured, and from a position of strength, if at all possible. The VC community is tiny and chatty.

## Investor updates and "potential investor" updates

I have found it useful to keep two mailing lists: One for potential investors, so that if you are introduced to a VC, you can tell them "we are not raising right now, but we have this mailing list where we send updates to interested potential investors, would you like to be on that"? This helps you stay fresh in their minds, and you can communicate important milestones and other good news via this mailing list.

The second list is the “investor updates” mailing list, where you can share more detailed information about the company, how it is doing, and even requests for introductions or hiring referrals. This list is a good way of keeping your investors informed, on your side, and excited about your progress, while also tapping into their network when it becomes useful.

Make sure that you post to the “investor updates” mailing list at least quarterly. Having good news to share is always a plus. And you can actually seriously anger your investors if you're not consistent with your updates.

The best “investor update” email lists I am on are fantastic at keeping investors in the loop, but also excited about the progress of your company, and able to chat among them about how good things are going. Excited investors that talk to other investors are going to be a big asset if you ever need to raise capital again.

## Picking the right VC

The VC claim of providing “value-add”

In the end, all VCs sell a homogeneous good: Money. In order to differentiate themselves, many funds now advertise that they provide various value-add services, from helping with customer calls, making introductions to potential customers, helping with hiring, making introductions to potential later-stage executive hires, etc.

These things are all nice on paper, but ... for many of these services, you may as well just take money at a higher valuation, and then spend it on service firms that do the job for you. (Be aware though that you **don't take money at a valuation you can't double in the next round**).

VCs **can** provide value-add, but it's often not where founders expect them to do it. Most importantly: VCs will not be able to help you build or run your business. That is your job, and a VC can't take over for you. **They are investors, not cofounders.**

The “big gorilla” theory of picking VCs

VCs can provide a lot of help when the going gets tough. They are experienced brawlers in a sense; their bread and butter is dealing with companies in crisis and snatching victories out of terrible tail spins. Just be aware that VCs will maximize their own return / minimize their own downside in times of crisis - they will be on your side as long as incentives are aligned.

I used to joke that taking VC is a bit like chaining yourself to a large silverback gorilla. If there is a brawl, you want to have the biggest and strongest gorilla on your side. On the flip side, you always need to have sufficient bananas at hand to keep the gorilla happy, lest he takes out his discontent on you. **The value-add of VCs is likely going to be greatest in times of crises that are unrelated to the core company-building.** Acquisition processes, conflicts, board strife, etc. In such scenarios, the value they can bring can be immense - clever GPs (“General Partners” - the folks that make the investment decisions at the VC firms) have been known to quadruple acquisition prices, pacify brawling boards and much more. It is these crises where VCs deliver the most value.

## Your relationship with the GP

In many ways you are picking GPs more than you are picking a VC firm. The GP is the person you will interact with, and the GP is the person with whom you need to get along and who you need to keep convinced that you are a winner. Make sure that you check the background of the GP (by talking to other founders with experience), and that you have a common vibe with the GP. A breakdown in the relationship between the founders and the GP is bad for everybody involved. **A good GP is a real asset** if used wisely.

“Why aren't you raising now?”

Some VCs have a tendency to ask “why are you not raising now” when you tell them that you're focused on building the product. Since you want to avoid an unintentional fundraise, tell the VC in a very friendly manner that you don't want to be distracted and need your foot on the gas. A friendly but firm line could be “we really do not want any distraction now, if you feel that strongly you can always send an unsolicited term sheet to the board to consider”.

## Early-stage valuation

Many first-time founders are confused by how valuation in early stages work. There are no comparables, there is no revenue, there is often literally just a promising story and a dream. How should things be valued? Both a “proper” priced round and SAFEs require some form of pricing.

The reality is that early-stage valuation is often more a function of “how capital intensive will it be to get to the next step”, and “how much equity can meaningfully be given up at this stage”. Often, the math comes out as follows:

1. It is customarily accepted that founders give up no more than 20% in a round.
2. If the company needs to spend 3m to get to the next stage, that implies that the valuation needs to be  $5 \times 3 = 15\text{m}$ .

Yes, this sounds very backward - because it implies that if you have a more capital-intensive business, you get a higher valuation. That is somewhat true - but it isn't “getting” a higher valuation, but “needing” a higher valuation.

There are exceptions to this. I don't know the details of CrowdStrike's first round, but rumor has it that they gave up **way more equity than the norm**, and used that to **raise much more money than the norm**. CrowdStrike was a special case, though - an extremely strong founding team, repeat founder CEO that had long-established trust relationships with the initial investors, a big established market full of slow Dinosaurs ready to be eaten, etc.

## Product and customers

Good product management is a surprisingly rare skill considering how important and largely common-sense-ish it is. **As a founder, you are also the initial product manager, and you better get good at this quickly.**

## Technology vs. product, starting from a problem vs. from a solution

The single most common way technical founders kneecap themselves is not **starting from a problem but from a solution**. Usually they have invented some form of technology that can do something new and impressive, and they are looking around for a problem that could be solved with it. "If all you have is a hammer, every problem looks like a nail" becomes a real problem. A founder with a technology in hand will tell themselves the most astonishing fairy tales about hypothetical problems that people might have that the technology is a solution for.

This is almost always the wrong way to approach the issue.

It is very easy to have **novel technology**. It is much harder to have a **novel product** that solves a **real problem**.

This problem is particularly prevalent with first-time founders trying to commercialize research done at university.

## Product ideation

At this point I should recommend the book "The Mom Test". This book is extremely useful, as it goes into depth about all the problems that can come from seeking **confirmation** vs. seeking **information** when talking to potential clients and users.

My recommendation for product ideation is: Have true, emphatic conversations with people about problems and pain points in their current problem space. Do not suggest a solution, just listen. Try to be the person they dump their (work) sorrows onto. Talk to great developers about what is creating friction in their workload, what pains them, talk to great SREs about what is and what isn't working.

Your job in these conversations is somewhere between therapist and intelligence officer: You want to elicit information that is not explicitly stated, you want to create a trust relationship so that the other party honestly shares their pain points, and you want to bond so that in the future when you have perhaps a product that might make their lives easier they will trust you enough to give it a shot.

Your goal is to get information about their employers business model (what drives their revenue growth, what drives their bottom, what drives customer satisfaction, what is currently blocking any of the above), their tech stack (what are their current processes like, what are they using, how adequate is it, etc.), their organisational setup (who makes decisions, how are these decisions implemented) - and then about the individual (are they well-placed to see the organisational problems, where in the value-creation do they sit, are they representative for a bigger cohort etc.).

Your goal is **also** to figure out whether you can help this person in a major way, by building a product that makes their life better.

## Top-line growth, bottom-line-growth, and everything else

Broadly speaking, there are three categories of products (and we will re-visit them when we talk about sales later):

1. **Top-line growth products.** The customer buys the product, and it helps them grow their revenue. These are the most popular products, and the best product category to be in.
2. **Bottom-line growth products.** The customer buys the product, and the product saves them money so that the bottom line is better even if the top-line is stagnant. While one would think this is as popular as #1, **it usually is not**, unless the buyer company is in trouble or the overall economy is in recession. Most economic phases are expansionary phases where gaining market share and growing revenue is more important than growing profitability.
3. **Everything else.** Security, compliance, etc. are all “cost of doing business”, but nobody actually wants to do them. These are cost centers, and the hardest products to sell.

Whenever you have an idea for a product, think about where in these buckets it falls. I generally do not recommend touching bucket 3 if you can avoid it, but that is a personal preference (more on this when discussing sales).

## The user persona and the buyer persona

When you are thinking about the product, there are two separate personas that you need to “create”: The user persona and the buyer persona.

**The user persona** is the person that will actually use the product. Think about that person. What are they like? Where in the organisation do they sit? What do they like? What content do they consume? What incentives help them advance their career?

In an idea scenario, you can draw a cartoon of the person, or create a fake on-paper LinkedIn profile of the typical user persona. Try having one persona, even though two or three may become realistic as your product evolves.

The user persona is the person into whose shoes you need to put yourself when making product decisions.

**The buyer persona** is the person that will make the decision that the product is to be purchased, and that the product is going to be installed / used. This is often not the user persona, and you need to do exactly the same steps that you did for the user persona here.

You need to be able to articulate: Who is the person that I need to convince to buy the product, and that has the ultimate say whether the product deployment becomes a success or a failure.

## How does this product get (user/buyer) promoted?

Both your user and your buyer need to see a way how buying and using your product will make them look good and potentially get them promoted. Getting their company to buy your

product is something where they need to fight for the product internally - so they need to take on personal and reputational risk inside the organisation. In general, people do not stick their necks out if they don't see a benefit from it.

You should figure out a good narrative for both your buyer and user persona how the product will improve some metric that will allow them to show within their organisation that their decision to push for the product was a good one, and that it helped the business.

## A shining example: The FinOps stunt

A fantastic example for the creation of an entire narrative and a new branch of the org chart came from the creation of the "FinOps" movement, which I watched from afar.

There was a time when "managing and optimizing cloud spend" was a bit of a stepchild that fell between different areas of an organisation: Finance was responsible for footing the bill, but ultimately it was operations / ops that made the technical decisions.

A company that was selling cloud cost management software found that it ended up with difficulties finding the right buyer persona inside of existing companies - should we sell to finance, or to ops? How do we do that?

Their answer was genius: They started producing all sorts of material that helped argue "between finance and ops there is a real gap, and cloud-native companies need to create a new team to bridge that gap -- and **you should be the leader** of that team". Their pitch to their constituency was essentially: Instead of moving up inside your department/lane, here is a lot of material with which you can argue that there should be a new department created, under **your visionary leadership**, that can then go and buy our product.

The result was the setting up of a foundation, certification, FinOps-centered conferences, branding etc. - all brilliantly and extremely cleanly executed. The company identified a real gap, but when faced with organisational hurdles that made selling their product difficult, they identified a plausible and clean path to fixing the problem by **creating incentives for people inside other organisations** to fix the issue.

## Development partners

When you are in the process of designing a new product, **having an early and continuous feedback cycle with real-world problems is absolutely crucial**. One of the best ways to achieve this is via **development partners**.

Development partners are organisations that fit your target market: They have the problem that you are solving, they have a user and a buyer persona, and **they are willing to provide feedback to you while you are building a solution for their pain point**.

Think about them as "initially non-paying customers that will help you get information about the real world as you're building your product". **Having several development partners** that are ideally quite different is absolutely crucial to building and refining your product.

Once you have decided on the problem you want to solve, and have fleshed out the user and buyer persona, finding development partners is one of the most important next steps. These arrangements can be extremely informal initially: Essentially just a coffee chat to discuss the problem. As you continue, things will become more structured, to the point where you can give the other party a prototype workflow to click through and observe their feedback.

## The importance of clickable prototypes

**Avoid building a lot of software without having gathered user feedback.** That sounds like a conundrum -- how are you going to gather user feedback without the software?

Prior to the advent of LLMs, the answer was "prototyping software that helps you build clickable prototypes". With the advent of LLMs, the answer may be "clever prompting", but in either case: It is important to get something that has a close resemblance with the final product into users hands as quickly as possible. Critical functionality can be missing / mocked out, but the goal is to de-risk what you are building before you have sunk a lot of resources into it.

## Design is not how it looks, Design is how it works

The above is a Steve Jobs quote that I find important.

Steve Jobs was pretty ruthless about the user experience, and a lot of Apple's product process was built on engineers preparing demos for **how something would look and feel** and then Jobs would be the curator of what got built. You're likely not Steve Jobs, but a monomaniacal focus on a great user experience is a trait we should all try to emulate. Fantastically engineered products with a great user experience actually make the world a better place - a world that is full of fewer items, but which work well, reliably, smoothly, and intuitively, is a better place.

## The "10 minutes to dopamine" rule

A great product will provide the user with a jolt of joy and dopamine within 10 minutes of touching it. If you can get below 10 minutes, even better. The first 10 minutes of the user experience need to engender a feeling of "wow, it's **that easy** to get going?" in the user, and then - quickly - begin fulfilling a (potentially small) part of the promise of the product.

At optimize, we were fanatical about minimizing the number of clicks and decisions the user had to make to deploy the software - "can we get this to three clicks?". On the flip side, after the Elastic acquisition, if a user wanted to set up continuous profiling, they had to go through about 50 decisions about the cluster size and exact configuration of the Elasticsearch cluster (that a normal person is not equipped to answer when they try out a product) in the Elastic cloud before they could start. This is an example of "what not to do".

## Be nice to your user

Remember that your product is there to provide value to the user. Keep the user workflows in mind, and be fanatical about being nice to the user: Do not force the user to make extra detours. Try to minimize the risk of accidents while empowering the user to do what they need to do. Help the user to get the maximum value in the minimum amount of effort.

No marketing dollar can buy you the same clout as a product where most users genuinely love using it.

I have seen products where - as I tried to use them - I asked myself "why do y'all hate your users so much?". Don't be that product.

Datadog's senior leadership is famously involved in every single click in their initial user interface, and changes to the important workflows go all the way to the top. This may not be right after a certain size, but for the foreseeable future of your company, it is right.

## Hiring

When your ambition outgrows what you can do alone, you need to hire. Hiring is hard, and you will make mistakes.

Hiring - which includes refining and disseminating a narrative for the company and where it goes - is one of the key things you need to do.

Great people like to work on great problems. So picking a fascinating market with interesting problems that also happens to be lucrative will make hiring great people easier. The best problems are so compelling that great people will gravitate toward your company. SpaceX is a great example - great talent wanted to work there early on just because the problem was inherently fascinating. For optimize, we had the great fortune that our message - "technically interesting, potentially lucrative, good for the environment while being pro-growth" - resonated with exceptionally talented people.

Having a great narrative for what you're doing and why is half the battle for hiring great people.

## Interviewing

My perspective on interviewing is that the interview process **should imitate the actual work that people will need to do**. I had seen at Google that whiteboard interviews have many terrible failure modes, so at optimize in 2019 we designed the interview process to be a take-home exercise that included writing a design doc, a round of feedback, implementation, and code review of the implementation -- essentially the entire roundtrip of what the day-to-day work would've looked like.

This worked stellarly well for everybody, and avoided many of the pitfalls of the whiteboard process. Most importantly: Take-home exercises avoid wasting time on candidates that are not serious or cannot complete the problem - a take-home exercise that is never submitted

does not eat into the interviewer's time in the same way that a whiteboard interview that fails does. When your time is at a premium, it's a fantastic filter.

## Managing people

There are libraries full of books about managing people. I don't think I can add too much here. The important lessons I have are:

1. In my experience, if you have hired well, you rarely need to motivate your reports. The better I got at hiring the more I ended up hiring people that I needed to motivate to take breaks and watch out for a healthy long-term balance.
2. People vary a lot. It is very difficult to fill a position exactly, often - especially in the early stages - it is better to hire somebody that is obviously smart, competent, driven, and a team player. As you figure out where their strengths lie, you tailor your organisation to maximize their positive impact. Big companies cannot do this, and one of the unfair advantages of a startup is that you can help people unleash their full potential.
3. You need a healthy amount of genuine curiosity and empathy to be an effective manager. These are things that can be practiced and improved. If you notice you just can't approach people's state from a position of empathy and curiosity, you may need to add someone who can.
4. Avoid inverse delegation (people pushing work that should rest on their level up their reporting chain), and instead try to provide support in solving the work on their level.
5. Be proactively curious about obstacles or pain points for maximum productivity. Work hard to know the sources of friction in your organisation and try to drive down drag. You're competing with very rich big players, you cannot afford not running a streamlined team.

## Conflicts in teams and the "capuchin monkey hierarchy"

When forming a team, one framework I've found useful is to think about the "Storming - Norming - Performing" phases of team composition: As you form a team, there is bound to be conflict in the team as everybody finds their role and niche. Humans are social animals, and everybody is status conscious to some extent. Conflicts often arise as the social construct of a team is settled and the implicit invisible hierarchy is established, and it is important to manage this in a way that these conflicts do not create lasting negative sentiment.

In some sense, teams are immutable: You add or remove a team member, and you no longer have the same team. Be prepared that any change in team structure will likely cause another cycle of "storming". This can be counterintuitive: Sometimes you have a team member that adds friction or drag to the point that you need to step in and remove / fire that team member. You'd expect that things will improve if you do that - but in general the removal even of a net-negative team member will cause new conflicts in the team, as the implicit hierarchy needs to re-settle.

## Having to fire people

Firing people is deeply unpleasant, and if you're a human with a normal disposition, you are likely going to think about firing somebody only when it's been painfully clear that they are not performing or harmful to the team, and you'll actually fire the person even later.

This is normal. We are taught from a young age to not arbitrarily exclude others from playing with us, and to have empathy with people that struggle. All of this is correct, and laudable. That said: You are also a sort of "leadership animal" in charge of the well-being of the entire "pack". And tolerating something that hurts everybody else is not living up to the trust the rest of the team has invested in you.

## Recognizing expertise where you don't have expertise

As a founder, you will find yourself in situations where you need to hire an external expert in a field that you know nothing about. This is a difficult game-theoretic problem. A friend of mine was diagnosed with cancer in his 20s, and the way he found the right surgeon was the following:

1. Find medical experts that have published on the relevant topic.
2. Ask them if they can take a meeting with you.
3. Ask them if they can rank the top-3 practitioners they would recommend in their field.
4. Intersect multiple of these lists to find people that are actually good.

A publication track record does not automatically imply practical expertise. It does imply that they know **something** about that field, and through knowing **something**, they will automatically have a much better instinct for ranking other experts than you do. In any field, the best practitioners are often not those that publish a lot, but the best practitioners tend to be known to people that do.

## Marketing

In the past, all my marketing was content-based: Write things that your target audience finds extremely useful. As they search for this information, they find you, and are immediately positively impressed. You enter a trust relationship with them by providing them with value first. "These folks have already provided value to me here, I wonder what else they can do for me?" is a good thought to induce in potential clients.

You already have a product story that you tell your user and buyer personas. A lot of marketing then becomes:

1. Catching people's interest.
2. Finding a good way of telling the product story.
3. Somehow managing to create a "dream" in the head of the potential client where they can imagine the product somehow helping them.

Now, the advent of LLMs and GenAI might have obsoleted some of the content-based marketing, but I think the core still stands: Provide value to build trust first, catch interest, tell

a story to create a dream, then enter the conversation. And once you're in conversation, you're moving from marketing to sales.

## Sales

For many engineers, and particular European ones, the word "Sales" is almost a dirty word. If you're going to start a company, you need to get rid of that attitude, and become serious about selling.

Interestingly, sales is one of the most quantifiable and quantified endeavours in most big companies, heavily tracked and pretty scientific in a broad sense.

### How does this sales stuff work?

Your marketing has built an initial amount of trust, told a story to create the germ of a dream, and then started a conversation. You're now in the realm of sales.

Sales work with a so called "multi-stage funnel". You divide your sales process into stages - from "initial contact" to "sale closed", or whatever works for your organisation. In-between these stages there are transitions - not every initial conversation will lead further.

The sales process then involves building mutual trust: You want to figure out whether the potential client is actually willing and able to eventually purchase something from you, and the potential client wants to figure out if you're a trustworthy company that can do something valuable for them.

### Managing a funnel

Each stage of the sales funnel has certain "hurdles" that have to be cleared to move to the next stage. You will learn that for clearing certain stages you need to provide certain pieces of collateral to the potential client (either to convince them, or so that they can convince other stakeholders on their side, or to perhaps provide them with a measurable success so they can obtain clout to advance the process internally).

Ideally, the person you're initially in contact with becomes your "internal champion" - the person inside the potential client organisation that will advocate for your product and try to get the product purchased. Without an "internal champion", there won't be a sale - someone on the inside of the organisation needs to push for the product. Most of the sales process is deepening the relationship with the internal champion and trying to empower them to do what they want to do.

### When to hire a sales professional

I have heard too many founders complain that their product is not selling and that they need to hire a sales professional to change that. **This is almost never true.** If you as the founder have difficulty selling the product, the most likely explanation is that the product isn't good

enough, or the story around it isn't good enough, or your user/buyer personas are not real enough.

Sales professionals cannot and will not fix any of the above. The sales process needs to work smoothly when performed by the founders before you should consider hiring a sales professional. That means that the first N big sales are the job of the founders. Only when you have a reasonable number of sales, and you have understood the full empirical sales process and refined the user and buyer personas, and you have a good story, and the product is good enough, and you have tested that this process is repeatable with essentially no changes on different clients -- only then is it time to hire a sales professional.

Once you have the full sales funnel, all the collateral you need on the way, and have executed the process multiple times, you get a sales professional to take all of the material and execute it repeatedly and scale it.

## Running the business

### Your three balance sheets

Your company ultimately has at least 3 balance sheets, one formal (money!), and two informal ones that you need to pay attention to nevertheless: Your technical debt balance sheet, and your personal energy balance sheet.

**Companies can fail by running out of money**, but if there's a serious-enough prospect of turning the company into success, you can always find investors / money to help you out with that.

**Companies can fail by getting too deep into tech debt** and hence crippling their own ability to innovate and react to changing customer needs, but you can often pay down tech debt by good hiring and giving good engineers a mandate and time to fix things. If your personal energy balance sheet goes too deeply into the negative, you're hosed.

**Companies can fail because the founders run out of energy. There's no investor that can pick you up from the floor if you've slid into burnout**, and you can't hire anyone else to truly run the company.

There's a jazz standard with the refrain "comes love ... nothing can be done". The same is true for burnout. **Your job as CEO or CTO or founder is to not run out of energy until the company is out of your hands**. For most founder types it is frightfully easy to lose track of this, and grind to a point where they do not function usefully any more.

Exhausted people make terrible decisions, from rash overreactions to slothful apathy. **Don't be an exhausted person.**

On the flip side: Your company will need all the energy you can sustainably put in, and then some. The work intensity will be much higher than in most dependent employment situations, because you are (in soccer terms) both the coach, the striker, and the goalie when the game gets close.

**The job of the founder is to run at maximum sustainable intensity. If you run with too low work input, the company is likely going to fail. If you run with too high work input and start malfunctioning as a result, the company is likely going to fail.**

**Be aware of your three balance sheets.**

## Cofounder relationships: The inverse marriage

Statistically, cofounder relationship breakdown is the #1 cause for startup failure. This is perhaps unsurprising: A cofounder relationship is among the closer relationships you'll have this side of joint parenthood. You're committing to invest many years of your life into a common cause, and committing to all the balancing that comes with it.

I sometimes joke that a cofounder relationship is an inverse marriage: When you get married, you do so in the hope that it'll last forever, and when a separation happens, there's usually a severe financial hit. You start a company with somebody in the hope that eventually you can separate again (and become richer in the process).

Managing the cofounder relationship is a real task. Doing it well involves a number of skills that many people only acquire through long experience or by being coached.

Some of the following advice will seem as if it came from books with advice given to young parents:

### Make room for non-operational conversations

For the company to succeed and all cofounders putting in their best, the company's success needs to be the one goal that all cofounders dream of and hope for. This means that all cofounders need to feel that their perspective matters, and that the trajectory of the company is right, and that conflicts can be resolved and repaired in a productive manner.

For this to work in the long run, it requires a fair amount of communication, often about emotional states, level of exhaustion, fears, goals, etc.

In therapy-speak this is (unfortunately) "being vulnerable". Personally, I think "open" would be a much better word. To quote Google's Gemini AI, "being vulnerable in conversation means openly sharing your thoughts, feelings and experiences, even if they expose you to potential judgement or rejection". To me, this is largely the same as being honest and transparent and making sure all important information is on the table. **Most people need to practice being open with each other because most of us fear social rejection a lot.**

In the stress of a growing startup, it is very easy for the founders to get completely eaten up by operational questions, and for there to be zero room for the sort of conversations that help align founders strategically. This can lead to a drift in goals and a form of estrangement between the founders, **which will eventually surface violently at inopportune moments.**

Make sure that the cofounders have regular time with each other outside of operational

chaos where they can discuss their own state, how they feel that things are going, and check whether goals or other sentiment is shifting. **Cofounders need to learn to be open with each other, and to trust each other that they can be open.**

One of the goals of this is that when push comes to shove, founders can have crucial conversations because they have practiced communicating with each other a lot. **A communication path that isn't exercised regularly will break down under stress, which is the worst moment.**

## Splitting equity between cofounder

Splitting equity between cofounders is both complex and simple. The simple part is: The right equity split is **where everybody feels that the split is fair.**

The more complex part is less about the initial split but rather about mitigating the risks for an eventual cofounder conflict or for an eventual departure of a subset of the cofounders. There are many mechanisms to consider here:

Sometimes investors want founders to "re-vest" some of their shares after a funding round. This is done in order to prevent founders from leaving quickly after a funding round (that might've included some secondary transaction).

There is also often "founder vesting", where all parties agree that the founder equity vests over time, so if a cofounder drops out early, they give part or all of their stock back.

A thing that both cofounders and VCs want to avoid, usually, is a cofounder leaving, no longer contributing to the company, but - by virtue of having an outsized stock allocation - benefiting from the hard work and grind that others put in after them.

There are many ways to skin this cat. At zynamics we eventually had an unusual construct that would freeze the company revenue at the time the cofounder left, and if the company ever exited, the exit revenue multiple would be applied to the revenue-at-time-of-leaving to calculate the cofounder stake.

The difficult part here is usually:

- 1) Having a difficult conversation about the consequences of break-ups and breakdowns of the cofounder relationship early on.
- 2) Figuring out a construct that addresses all concerns while maintaining good incentives.
- 3) Trying to make this compatible with the tax law in each country.

Number 3 is usually the hardest, as you can run into bizarre issues such as founder re-vesting creating very unfavorable tax treatment etc.

If you cannot come to an agreement that all cofounders think is fair, appropriate, and manages the risks properly, then you have the wrong set of cofounders.

## Relationship between founders and investors

The next common reason for company failure is a breakdown of the relationship between founders and investors. This can happen for many reasons, but in the end almost all relationship breakdowns are **matters of trust**. And given that the incentives for investors and founders can easily get out of alignment in times of stress, trust breakdown can easily happen.

Most of us have a definition of trust in mind that involves being able to rely on the other when things are going badly. This is likely not the right definition of trust to place in professional investors, because investors have a legal obligation to their limited partners more than to the relationship with you. This is not malicious, just a reality.

My advice is to be very clear-eyed that investors will have their own best interest in mind at all times. If you stick to this principle, trust is less likely to erode, unless the investor does something that is clearly harmful to themselves (which is hopefully unlikely). Assume all sorts of maneuvering, and before you get angry about that, just remind yourself: The nature of investors is that they will try to maximize their return. Their nature is not to be a trusted harbor in a storm. **You can be grateful when an investor does you a favor, but never be disappointed if they do not.**

## Consider getting a coach

Running a startup is hard, similar to trying to play sports at a high level. Very few high-level athletes can perform well without a coach (although there are a few!). Check if it's helpful to get a coach for the journey: A person with whom you can discuss the inevitable psychological strain that the journey will place on you.

Coaches and therapists exist in many colors. Curiously, what has been best for me was a therapist that mostly repeats what I just said with a sarcasm-laden intonation to make me reflect on what I am thinking - but everybody is different. It is worth trying to find someone that fits your personality well. A startup journey is a high-stress endeavour that will bring out the best and worst in your personality, and it is very helpful to have external reflection. **"Don't believe everything you think"** is good advice, but without an external perspective it can be hard to do that.

You might be lucky to have friends, spouses, or family members that happily fill this role for you - I am not saying you need to pay someone to do it - but you will likely benefit from having someone reflect on your thoughts & state with you.

## You and your employees

Managing the relationship to your employees is another important facet. Again, learning how to communicate openly is very helpful. For most employees (who are always in a somewhat "weak" position with regard to the CEO) it is crucially important that they can trust their bosses and CEOs to have their best interest in mind (where collective "their" means "all employees"). My advice (but also my leadership style) is "be open, be trustworthy, have the best interest of the entire team in mind, and facilitate bottom-up information flow".

One thing I am always terrified of is that important information does not get transmitted up the chain to me. As a result, I usually work extra-hard to make sure people always feel it's OK to bring me bad news.

Beware, though, of "inverse delegation" - sometimes people can mistake your openness to bad news as an invitation to bring you problems that are theirs to solve. Recognize this, and make sure problems get solved at the appropriate level.

## Equity and ownership with employees

Each employee will have a different risk appetite and prefer a different split between equity and salary. Some cultural aspects come into play here - a lot of European folks seem to not value equity as highly as most US folks do - but by-and-large, figure out what the employee wants and needs, and then construct a package that makes everybody happy.

For equity, you do want to make sure there is vesting - industry-standard one-year cliffs are generally a good idea. Having more equity can - sometimes - instill more sense of ownership and responsibility, but this is variable between individuals, so it is something to discuss - and to try. If you're in Europe, be extra careful about the tax implications of employee equity in each jurisdiction. What you want from an equity plan is that the employees are taxed on liquidity, not on grant or vest. There is almost always a way to construct this, but it unfortunately requires a per-jurisdiction approach. Talk to a good lawyer (which is not always an expensive law firm, see the note on "how to find experts") for each jurisdiction, there are subtle nuances that need to be reflected in the equity granting paperwork to make this work.

## Delegating work and personal growth

A founder's job for the first years of the company is one where you constantly need to learn something new, then work hard to find someone else to do it better than you, and repeatedly work on putting yourself out of a job.

This means you're pretty constantly trying to delegate what you're doing, and growing into something new. If you find yourself doing the same thing repeatedly, you should ask yourself: How can I delegate parts of this.

What you won't be able to delegate as cofounder or CEO is: Repeating and refining the story of the company over and over again, setting and repeating the company culture, and interacting with the investors.

## The end, the exit, the denouement

For most entrepreneurs, the journey will eventually come to an end - perhaps after 4 years, perhaps after 10, perhaps after 15 and a long after an IPO. You might be an exception, but the majority of companies are created, grown, and scaled with the intent of eventually letting go.

There are sections to write here, but they will have to be written for version 0.2 of this document, so not today.