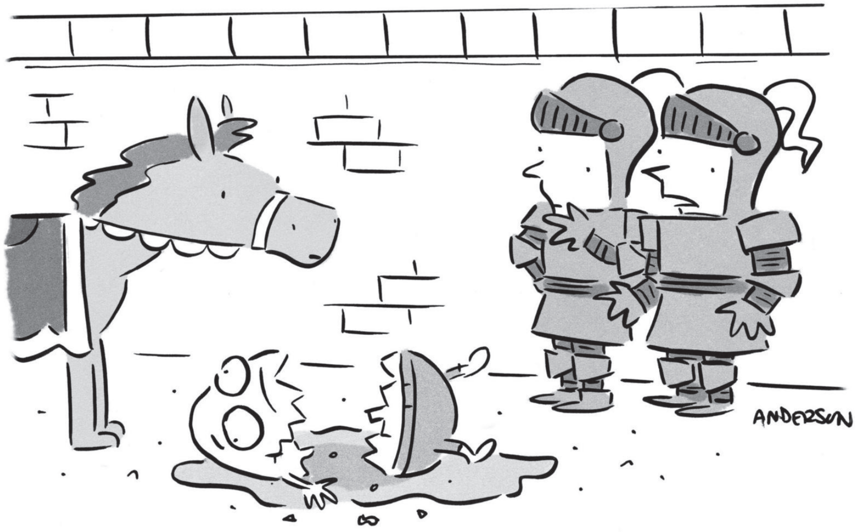
Brittle Points: How to make companies robust

Companies fail for all sorts of reasons.[2] Sometimes we're not surprised, like if the product never worked or not enough people wanted to pay to solve the problem in question.

But sometimes the product did work, people were paying to solve the problem, sales were starting to pick up, and then something broke, and it was all over. How does that happen?

- The product was built on a platform,[3] and the platform changed. A popular app drops to zero downloads after Apple builds it into iOS; a Google Workspace add-on drops to zero sales after Google builds that feature into Docs; a Twitter management tool breaks when Twitter removes functionality from their public API.
- The initial marketing channel quickly saturated,[4] so growth stalled[5] at a non-zero but unsustainably-low rate.



"I'm less interested in putting him back together, and more concerned about the giant chicken who put him up there."

credit[1]

- The initial marketing channel was sustainable for a while,[6] but got wiped by external forces: large bidders tripled the cost per click,[7] Google's SEO algorithm changed, the big industry event stopped happening, the link-sharing site became irrelevant, the hot blog lost its traffic, the magazine running the ads finally closed.
- One big customer,[8] representing 80% of total revenue, left. It wasn't a mistake to sign that customer—it funded the company.
- A key employee left the company. Early on, a 10x person[9] can mint the company but also could be irreplaceable. A suitable replacement is too rare; it takes too long to find someone, convince them to join[10] for almost no salary, and get them up-to-speed and productive.

I call these "brittle points"—places where sudden changes mean the company catastrophically fails, regardless how wonderful everything else is.

All young companies—and some mature ones—have dependencies like these. You can't help it; you have to rely on other technology to build the product, services from vendors to deliver the product, and human beings to do work.

Engineers have Brittle Points in their infrastructure, so they've developed common patterns to address it. Let's briefly look in how they do that, because it will give us clues about how to solve the problems in that list above.

A BRITTLE SERVER

Suppose we have a single server that runs our website. Any number of things can cause this server to break—a power failure, a network failure, a bad configuration change, too much traffic arriving at once, bugs in the code, all sorts of things.

How do we make this system less brittle?

Consider power failure. Power can fail if the power supply* inside the server burns up, or the power strip fails, or the power cord fails (maybe through a wetware[11]) failure like accidentally unplugging it).

We can address this Brittle Point with a second copy of the power components—a second power strip with a second cord plugged into a second power supply. This is, in fact, exactly what data centers do!

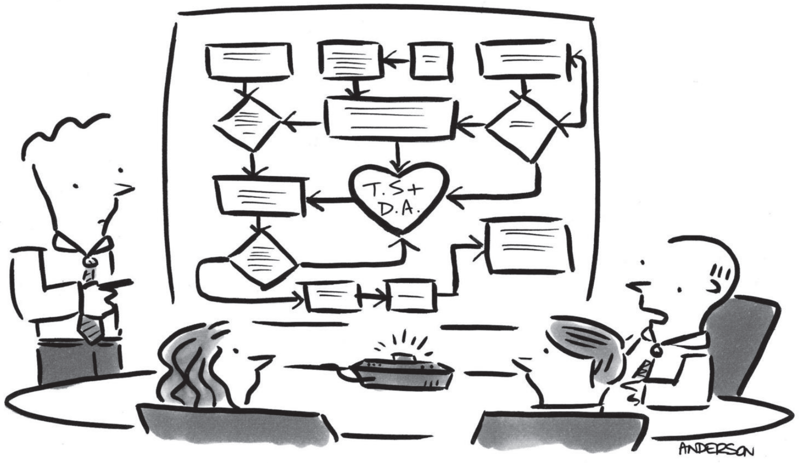
In short: redundancy—having two things that do the same job. **It's twice as expensive, but it buys robustness.** This is also what airplanes do.[12]

But what happens if the city power fails? Data centers have their own gas-powered generators. Which means they stockpile gasoline. Rarely-used gas-powered engines tend to fail, so they also test and maintain those units weekly. Data centers often have multiple generators. More robustness, purchased at significant, on-going expense.

In modern clouds we go yet another step further, because the entire data center is itself another Brittle Point. So we have additional, identical servers in a physically-separate data center, that draws power and network connectivity from different outside vendors. But now we also need a smart networking system that knows how to direct internet traffic to only the servers that are currently available. Yet another system, which itself becomes—you guessed it—a Brittle Point.

The pattern continues—fix more Brittle Points, at more cost and complexity, sometimes creating new Brittle Points. Reliability is expensive.

This pattern is applicable to all of the causes of failure above.



"OK, but what happens to the supply chain if you and Debbie break up?"

credit[13]

* This is the component inside the computer that receives the power cord; it converts city-power into the type of power needed by the other components.

NEUTRALIZING BRITTLE POINTS

“One platform” is brittle, because if the platform-owner forward-integrates (i.e. copies you), or removes APIs that you depend on, or themselves fail, that’s the end of your company. One solution is to be multi-platform.* Another solution is to only build on platforms where you have a high degree of confidence that the platform owners are committed to supporting their ecosystem by never directly competing with them. Ideally, the platform even promotes their ecosystem, so that it becomes a growth vector instead of a Brittle Point. (Salesforce is currently the best in the world at this.)

“One marketing channel” is brittle, because if anything happens to the channel, that could be the end of an otherwise-healthy company. The solution is to find additional marketing channels,[14] so that variation in any one of them is not fatal. Of course this also creates growth; again this “double-win” of “growth + risk-mitigation” shows why it’s especially valuable to invest in.

“One big customer” is brittle. One solution is a long-term contract with a serious breakup clause—insurance to bridge the time it will take to win replacement customers. Another solution is to prioritize sales until that customer represents a smaller percentage of revenue. Another is up-front payments,[15] so you have the cash-flow to invest in that growth. Another is to charge even more than you originally calculated, again for mitigation cash-flow. The typical attitude is, “We now have a large customer, so pour extra money into development to make sure we don’t lose it,” but the right attitude is to use that money to win other customers.

“One key employee” is brittle. Not only might they leave, but they will inevitably get sick or take vacation. The usual refrain in the startup world is that none of these are options—everyone has to work 70+ hours/week[16] to the exclusion of other things.[17] Talk about brittle!

* Example: At WP Engine, we run on all three major clouds—AWS, Google, Azure. Example: A marketing tool for listening and responding on Twitter could add support for LinkedIn, Threads, and Bluesky. In this way, additional features can also be risk-mitigation.

Solving these things takes time and money. Like the server example, they’re not free, and not quick-fixes. You can’t just hire three more fantastic developers to create a robust engineering team, and you can’t just snap your fingers and find three new efficient, productive marketing channels.

Therefore, the right attitude is to **maintain a list of these risks** and then periodically ask:[18] **Which single one is best to attack right now?**

For example, it’s cheaper and easier to experiment with new marketing channels than it is to find, interview, convince, and manage a second software developer, and plus if you can get a second marketing channel online, that will generate revenue, which in turn means you can *afford* a second software developer. In this mini-scenario, the best thing is to focus all your energy on getting a second marketing channel working.

As you scale up[19] **the size of the “chunks” that create brittleness also scale up**, which creates new brittle things, and thus new risks and new investments. For example, with $260M in revenue in 2016, still growing at a blistering 60%/year, with a thousand employees, Hubspot was not brittle in any of the ways outlined above. But they recognized that they were a single-product company. At that scale, that’s a Brittle Point: if there were a sea-change in the market for inbound marketing software, that could be fatal to Hubspot. It also limits long-term growth[5] as the market matures and saturates. The way out—the redundancy—was to become a multi-product company. Furthermore, the second product had to scale at least as well as the first one; it’s still a Brittle Point if one product is 95% of revenue. They attacked that problem, and in 2024 it’s clear that they succeeded.*

* See this article[20] for data and discussion.

THE BIGGEST BRITTLE POINT

Finally, on a personal note, there's another "chunk-level" that's even larger than all of the preceding, and **it's a brittleness that almost all founders suffer from, including myself**. The chunk of "the entire company."

This is a one reason why founders are almost always sad and sometimes permanently depressed after a successful sale of a company.[21] This was your identity, your life, for years. You don't remember what it was like to "be you" without it, and anyway "you" aren't the same "you" anymore. You don't have hobbies or even good friends anymore. You might have sacrificed family or health.[16] Talk about a Brittle Point. Your entire life is a Brittle Point.

The solution here is not to have two companies or two jobs. That's burnout; a lack of singular focus creates worse outcomes.[17]

Rather, the solution is to realize that there were things you did and loved before and there will be things you will do and love after. They might not all be the same things. Sometimes it's best if they're not; you're a different person now. You are inside a chapter in the book of your life. Even if one chapter is sad or has an unexpected twist, there's the next chapter which you can look forward to, even if you don't yet know how that story will unfold.

You can rediscover who you are with this process.[22] But it's rarely easy, or simple, or fast. Hopefully the successful sale has literally bought you time.

Robustness, not in many things simultaneously, but in things serially. That's what you do with limited time, and how you navigate the arc of your life's story.

Back to today and the here-and-now. Go list all the Brittle Things you have today. Then tackle one or two of those things at a time—you have to manage risk,[23] not try to eliminate all of it at once.

Be thoughtful, and build steadily away from brittleness.

The current version of this article:
`https://asmartbear.com/brittle-points/`

More articles & socials:
`https://asmartbear.com`

References

1. https://andertoons.com/egg/cartoon/6898/not-interested-putting-together-concerned-about-giant-chicken
2. https://longform.asmartbear.com/avoid-blundering/
3. https://longform.asmartbear.com/marketing-platform-independence/
4. https://longform.asmartbear.com/low-budget-marketing/
5. https://longform.asmartbear.com/growth-asymptote/
6. https://longform.asmartbear.com/one-channel/
7. https://longform.asmartbear.com/bootstrapped-cpc/
8. https://blog.asmartbear.com/find-first-big-customer.html
9. https://longform.asmartbear.com/imbalanced-people/
10. https://blog.asmartbear.com/startup-hiring-advice.html
11. https://en.wikipedia.org/wiki/Wetware_(brain
12. https://longform.asmartbear.com/whos-lying/
13. https://andertoons.com/love/cartoon/6341/what-happens-to-the-supply-chain-if-you-and-debbie-break-up
14. https://longform.asmartbear.com/exponential-growth/
15. https://longform.asmartbear.com/annual-prepay/
16. https://longform.asmartbear.com/sacrifice-your-health-for-your-startup/
17. https://longform.asmartbear.com/two-big-things/
18. https://longform.asmartbear.com/strategic-planning/
19. https://longform.asmartbear.com/scale/
20. https://longform.asmartbear.com/product-market-fit/
21. https://longform.asmartbear.com/identity-selling-sadness/
22. https://longform.asmartbear.com/pivot-points/
23. https://longform.asmartbear.com/startup-drake-equation/