

The Important Thing — powerful enough to override all your deficiencies



"I don't care *whose* time it is, we're going to have a nice dinner for once!"

credit 1

Do you feel the crushing weight of the disadvantages facing every new company? No brand, no features, no customers, no money, no distribution, no search engine rankings, no efficient advertising, no incredible executive team, no NPS, no strategy.

How do the successful startups win anyway? Do they solve all those problems at once, or at least quickly?

No.

One answer is they pick their battles where they have advantages over incumbents.²

The other answer is that they need one thing to go really *really* right, to overcome the torrent of things that are going wrong.

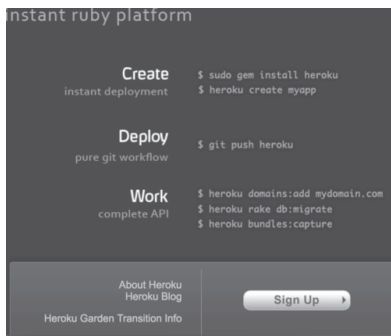
I apologize in advance for using the dang iPhone as an example but... The iPhone is one of the most successful and important products of the past few decades. But the first version launched with a mountain of issues. It was a terrible phone, ironically. The whole idea was that it was a "smart *phone*," yet everyone agreed their cheap-o Nokia flip-phone was ten times better at being, you know, a *phone*. Also, imagine launching an operating system that didn't include "copy/paste." Terrible!



credit 3

But, the iPhone did something so well, that people wanted so badly, they would put up with all the other

crap: You could use the internet. The *real* Internet with full websites and everything. The web actually *worked* (even if slowly). Email actually worked. In your pocket. It's hard to explain the magic and excitement to a Gen-Z'er who takes it for granted. This was so compelling, all the other problems didn't matter.



For more than ten years—an eon in tech—Heroku has been the dominate way that Ruby on Rails developers launch public applications. When it first came out, it was rife with “deal-breakers” that developers continually winged about. “What do you mean I have to use Bundler—it’s broken half the time!” “What do you mean I can’t change the filesystem at run-time—

I’ll have to change my algorithms!” “What do you mean it doesn’t support MySQL—everyone uses MySQL! My queries are going to break.” “Wow these websites are really slow.” On and on with the complaints, and all quite valid.

But, Heroku did something so well, that people wanted to badly, they would put up with all the other crap. You could type `git push production` and your site went live. You could use a knob on a web page to determine how scalable the site was. (Don’t worry, that knob is also connected to your wallet.) You never saw a server. You never thought about backups. You never worried how to securely stash your API keys. You always had a staging area to test things in a real server environment before pushing code live. DevOps became a thing of the past for a large class of applications. This was a revolution so important, so compelling, all the

other problems didn’t matter. Developers changed their workflows and their code around Heroku and “12-Factor”⁴ apps; Heroku did not change to suit developers.

This is a universal pattern I call **The Important Thing**.

Break-out products deliver something so fantastic, so game-changing, so important to the customer, that this one thing is sufficient to override the otherwise-overwhelming deficiencies of the rest of the product and company. So great that people tell their friends or force their colleagues to use it too (defeating the lack of marketing). So great that they’ll use it even if support is slow and releases have bugs (defeating the lack of operational excellence). So great that they’re excited to support a promising new company instead of worried about creating a dependency on a wobbly new company (defeating the lack of brand).

In short, despite the startup not having the positive attributes of large companies (brand, service, features, stability, integrations, social proof), the startup can still win² because their Important Thing is so compelling.

The Important Thing isn’t always a feature or technology. Fogbugz was never the leading bug-tracking system, but I employed it for most of the 2000s because I was a huge fan of Joel Spolsky’s blog,⁵ so it felt good to use a product made by a company whose values and behaviors I respected and learned from. The same happened with Basecamp and the 37signals blog.^{6*}

Here’s how to apply this to your own business:

* You might be tempted to say Basecamp was successful because the work-style espoused by 37signals leads to successful products, but the contrary evidence is that all of their many subsequent products—built with the same work-style, by the same people, and even with the same code base—were all dramatically less successful, to the point that all of them have now been discontinued, and the company has been renamed “Basecamp” to emphasize that only the first of those experiments was ultimately successful. *Editor’s Note:* Now in 2024, they seem to be successful with hey.com.

It's easy to get overwhelmed by the myriad of inadequacies you undoubtedly have. It's tempting to attack them all, but worrying about everything and attacking simultaneously on all fronts with no weapons just leads to burn-out, and does not result in a company that is excellent on any front. Fortunately, you don't need to solve all those problems. You need to solve almost none of them.

Instead, you need to focus on the one thing (maybe two) which is your Important Thing. The thing where, if you're extraordinarily good at it, customers will overlook everything else.

It could be a feature (e.g. disappearing messages with Snapchat), but you can look beyond features. It could be enabling a lifestyle (such as remote-work or with-kids-work). It could be your online reputation (e.g. Joel Spolsky for me). It could be that you're solving a problem in an industry that others overlook; having "any solution, even with problems" is better than having no solution. It could be that your culture resonates with an audience (e.g. 37signals), maybe due to an informal voice in an otherwise formal market, or because you have a cause—a higher purpose—so that people aren't just buying a shirt or some software but rather they are supporting a movement (e.g. Patagonia who cares so much about the environment that there's a company policy that they will bail employees out of jail if they're arrested for peaceful protest, and by the way one of the results is that they have only 4% annual employee turn-over).

You should select something that you *want* to obsess over for the next five to ten years, that gets your customers excited, and which you at least have a possibility of executing. And then do that. Maybe *only* that. If you let all the other fires burn, maybe you have a shot at actually being excellent at that Important Thing.

If you do, all those other disadvantages exist, but aren't fatal. That's all you can hope for, at the beginning.⁷

And all you need.



"Any way we can hurry this up?"

credit: 8

The current version of this article:

<https://books.asmartbear.com/anthology/the-important-thing/>

More articles & socials:

<https://asmartbear.com>

© 2019 A Smart Bear Press

REFERENCES

1. <https://andertoons.com/responsibility/cartoon/7817/dont-care-whose-time-it-is-were-going-to-have-a-nice-dinner>
2. <https://longform.asmartbear.com/startup-beats-incumbent/>
3. <https://www.mic.com/articles/164769/how-big-was-the-first-i-phone-how-the-i-phone-s-screen-size-has-changed-over-time>
4. <https://www.12factor.net/>
5. <https://longform.asmartbear.com/letters-to-joel-spolsky/>
6. <https://m.signalvnoise.com/>
7. <https://longform.asmartbear.com/chaos-at-start/>
8. <https://andertoons.com/death/cartoon/1705/any-way-we-can-hurry-this-up>