Rare things become common at scale

"The idea is to not only grow our piece of the pie, but to expand into other nearby pastries."

Something interesting happens when you run more than 1,000 servers, as we do at WP Engine,* powering hundreds of thousands of websites.

Suppose that on average a server experiences one fatal failure every three years. The kernel panics (the Linux equivalent of the Blue Screen of Death), or both the main and redundant power supply fails, or some other rare event that causes outage. This isn't a quality issue—this is normal. This isn't something to "fix."

> " *Windows NT crashed.*
> *I am the Blue Screen of Death.*
> *No one hears your screams.*"
> —*Haiku from FSF*[2]

But remember, we have 1,000 servers. Three years is about 1,000 days. So that means, on average, **every single day we have a fatal server error.**

Not to mention 10 minor incidents with degraded performance, or a DDoS attack somewhere in the data center affecting our network traffic, or some other thing that sets pagers a-buzzing in our DevOps team and mobilizes our Customer Support team to notify and help customers.

"Well sure," you say, "that's normal as you grow. If you had just 10 servers and 100 customers, you'd have fewer problems and many fewer employees. Today you have more customers, more servers, and more employees. What's so hard about that?"

The insight is that **scale causes rare events to become common.** Things happen with 2000 servers that you *never saw even once* with 50 servers, and things which used to happen once in a blue moon, where a shrug and

---

* Editor's Note: As of 2024, this article is ten years old; we now run twenty times as many servers, and the lesson of this article continues to be accurate.

"Boy, what a day, huh?

credit³

a manual reboot every six months was in fact an appropriate "process," now happen every week, or even every day.

Things as rare as, well, you know…

It's not only *problems* that morph with scale, but **your ability to handle problems**.

For example, a dozen minor and major events every day means 20-50 customers affected every day. Now consider what happens as we try to inform 50 customers. For some we won't have current email addresses, so they don't get notified. Some of *those* will notice the problem and create extra customer support load; at worst they'll post on Twitter about how their website was slow or offline today and WP Engine "didn't even know it." Then our social media team has to piece all this together, attempt to respond, maybe put together a special phone call with that customer, and so on. Those customers are also more likely to leave a bad review on some review site, compared with the 99.99% of customers who experience no

such incident, but also had no reason to decide that "today is the day I will go to a review site and leave a good review."

Or consider the scale-ramifications of on-boarding 1,000 new customers a month. In that case, it's likely that any given server issue will affect a customer who has only been with us for a month or two. Thus the issue causes a "bad first impression," which is harder to address than a customer who has been with us for three years and has built up a bank account of patience.

So, rare things being common isn't just difficult from the operational side, but also when you try to handle those problems with customers or other downstream consequences, causing *much* more work to solve than when the company was small.

The usual response to this is "automate everything."

As with most knee-jerk responses, there's truth in it, but it's not the whole story.

Sure, without automated monitoring we'd be blind, and without automated problem-solving we'd be overwhelmed. So yes, "automate everything."

But some things you can't automate. You can't "automate" a knowledgable, friendly customer support team. You can't "automate" responding to a complaint on social media. You can't "automate" the recruiting, training, rapport, culture, and downright *caring* of teams of human beings who are awake 24/7/365, with skills ranging from multi-tasking on support chat to communicating clearly and professionally over the phone to logging into servers and identifying and fixing issues as fast as (humanly?) possible.

And you can't "automate" away the rare things, even the technical ones. By their nature they're difficult to define, hence difficult to monitor, and difficult to repair without the forensic skills of a human engineer.

Does this mean all our customers have a worse experience? No, just the opposite. Any *one* customer of ours has *fewer* problems per year now than a year ago, because we're constantly improving our processes, automation, hardware, and human service. It's when you look across the entire

company, and the non-linear additional effort it takes to not just improve the average experience, but to manage the worst-case experience, that you appreciate the difficulties.

This explains the common effect where people complain about a company every day on Twitter, yet you yourself have never had an issue with them. The paradox is solved by realizing that "rare things" means *you* probably never experienced it, but at scale, *someone* is experiencing it each day.

Does that give high-scale companies like WP Engine an excuse to have problems? No way! In fact, if we're not constantly improving on all fronts, the scale will catch up and overtake us.

But for those of you in the earlier stages of your companies, when you project 5x growth against 5x costs (or only 3x the costs because you'll get cost-savings at scale), you're guessing low. When you show 5x growth in projections but don't budget for new hires in areas like security, technical automation, specialized customer service areas, and managers and executives who have trod this path before and come battle-hardened with playbooks on how to tackle all this, you're heading for an ugly surprise.

And with high growth, the surprise appears quickly, and recovery means acting twice as fast again to claw out of the hole and then finally get ahead of it.

Scaling is hard![4]

Current version of this article:

https://longform.asmartbear.com/scale-rare/

More articles & socials:

https://asmartbear.com

REFERENCES

1. https://andertoons.com/selling/cartoon/6497/idea-not-only-grow-piece-of-pie-but-expand-into-other-pastries
2. https://www.gnu.org/fun/jokes/error-haiku.html
3. https://andertoons.com/devil/cartoon/6064/boy-what-day-huh
4. https://longform.asmartbear.com/scale/