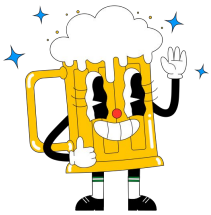
Principled identification of "root causes" using techniques from safety engineering

Laura de Vesine
Datadog, Inc.

A QA Engineer Walks into a Bar

Orders some ridiculous things



First real customer walks into the bar and asks where the bathroom is

I've been told to start talks with a joke, so here's a joke that I originally saw on Twitter.
A QA engineer walks into a bar. Orders a beer. Orders null. Orders -1 beers. Orders 999999999 beers. Orders a lizard. <click>
First real customer walks into the bar, asks where the bathroom is… and the bar catches on and everyone dies

Original joke from Twitter; including but not limited to
https://twitter.com/brenankeller/status/1068615953989087232?lang=en
Lizard stickers created by Stickers - Flaticon
Beer stickers created by Stickers - Flaticon
Number stickers created by Stickers - Flaticon
Bar Photo by Oliver Frsh on Unsplash
Fire Clipart from https://pixabay.com/users/clker-free-vector-images-3736/

Whoops

- Write a postmortem
- Examine the incident timeline
- Cause: Customer needed directions to the bathroom
- Solution: add a sign pointing to the bathroom

Well that's not good. But we're SREs! Let's do some root cause analysis, create a postmortem, and fix our system so that doesn't happen again. Taking a look at the timeline of the incident, it's clear that what went wrong is that a customer asked where the bathroom is. We can resolve that issue by making sure customers don't need to ask where the bathroom is, and we can do that by adding a sign. (short pause)

Problem solved! We've done all the right things – analyzed our incident, determined the root cause, and fixed it. We even wrote a document on how we could move the bathroom to an easier-to-find location, though we decided not to prioritize that for this quarter. And we'll get to internationalizing the sign eventually too. (pause)

Washroom Sign Photo by Prateek Katyal on Unsplash

So, time keeps chugging along and our bar business grows. We get more customers, who order more drinks, and we make more money (woohoo!). Sometimes, our bar is busy enough that the bartender gets behind for a little while, but they always catch up after a few minutes. That is, until one evening when there's even more customers than usual, and two customers try to order at the same time. When that happens <click> the bar catches on fire and everyone dies

Flame Image by Monika Grafik from Pixabay
Bar Photo by Victor Clime on Unsplash

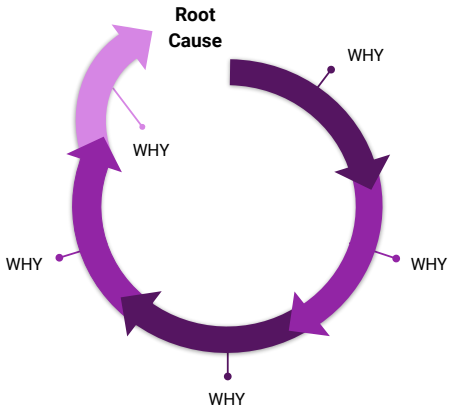
Let's Use "5 Whys"

Ask "Why?"

- exactly five times
- to find exactly one root cause.

Be blameless

Use differs in practice



Hm. We seem to have an unreliable bar, and our previous method of finding the root cause didn't do the trick for keeping the bar from burning down. Our boss is unhappy with the results, and "suggests" we use a formal analysis method like "5 whys".
Like any good engineers, we get on the internet to figure out how to do a "5 whys" analysis. The Wikipedia article on 5 whys analysis tells us that the technique (as originally laid out) is to ask "why" exactly five times, to find exactly one root cause.
Now, I realize you might have some criticisms of that but for our bar let's go ahead and use the "original" method

5 whys definition from wikipedia: https://en.wikipedia.org/wiki/Five_whys

Asking “Why?”

1. Why did the bar burn down?
   Because the bartender got distracted
2. Why did the bartender get distracted?
   Because there were too many customers ordering at once
3. Why were so many customers ordering at once?
   Because the bartender couldn’t keep up with demand

So let’s ask why. First obviously we want to know why the bar burned down – it’s because our bartender got distracted. We want to stay blameless, and it’s the natural next question, so we ask why our bartender got distracted. We know that’s because there were a lot of customers ordering at once, and that that happened because we had a demand backlog

Question stickers created by Stickers

5 Whys (Continued)

4. Why couldn't the bartender keep up with demand?

   Because the bar is more popular than it used to be and there were not enough bartenders

5. Why were there not enough bartenders?

   Because customer traffic has surges, and we didn't have some way to rapidly increase bartender capacity

Why'd we get a demand backlog? Well, obviously because we had a capacity shortfall. And we had a capacity shortfall in this case because we need to be able to add more bartenders on short notice as we get surges of customer traffic.

Question stickers created by Stickers

Root Cause Identified!

We can add “flex” bartender capacity

Let’s build an app for that

Everyone gets promoted

Problem solved

And voila! We’ve identified our root cause!
Let’s make our overall “bar” service more resilient by coming up with a way to schedule “surge” bartenders, maybe only pay them when they’re actively working. (short pause) We can build a phone app for it and leverage the gig economy! It’ll be cool and Web 3.0 and we’ll all get promoted for solving the technically complex problem (with buzzwords!) of how to only pay a bartender when they’re actively pouring drinks for people (oh, and also our bar now has elastic capacity, so it’s more reliable)

Phone Photo by Neil Soni on Unsplash
Star icons created by Freepik - Flaticon

Narrator: the problem was not solved

Some time goes by and the bar doesn't burn down…

Until one night the bartender leaves the heat on overnight

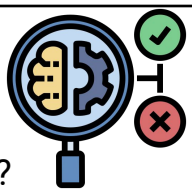
The bar catches on fire

This time, no one dies!

<click> So, some time goes by and our bar doesn't burn down for a while… but as you've probably guessed, one night something new happens: in this case <click>, our bartender leaves the heat on overnight. And, <click> the bar catches on fire. Good news! this time <click> it was the middle of the night, and no one died.
But it's important to analyze our "near misses" too (I'll actually talk more about how and why to frame that in a moment) – our customers *could* have been impacted, and we seem to have just gotten lucky with timing this time. (short pause)

Success icons created by Freepik - Flaticon
Thermostat icons created by Iconjam - Flaticon
Fire icons created by Freepik - Flaticon

5 Whys Redux

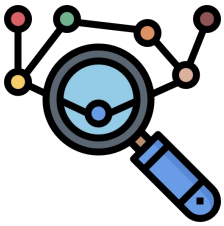


1. Why did the bar burn down?

   Because the heat was left on overnight
2. Why was the heat left on overnight?

   Because it was a record cold night and the bartender didn't want to freeze to death in the morning
3. Why was it a record cold temperature?

   Because of global climate change



Well, we have this "5 whys" tool; let's try that again, maybe trying really hard to dig deeper into the big causes of what's going wrong. Why'd the bar burn down? Because the heat was left on overnight. Why'd the bartender leave the heat on? Remember to be blameless here… because they didn't want to freeze to death in the morning, and it was a record cold night. Why was it a record cold night? Well, because of climate change.

Question stickers created by Stickers - Flaticon
Scientist icons created by surang - Flaticon
Investigate icons created by noomtah - Flaticon

5 Whys Redux (Continued)

4. Why is there global climate change?

   Because of the human use of fossil fuels

5. Why are humans using substantial amounts of fossil fuels?

Capitalism

A-ha! So we can keep our bar from burning down by ending capitalism!

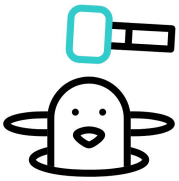
…That doesn't seem right

Why is climate change happening? Well, the scientific consensus is that it's caused by human use of fossil fuels, and (at a high level) we use a lot fossil fuels because of the pressure of capitalism. <click> Awesome! So our root cause is capitalism, and we can keep our bar from burning down by ending capitalism! (pause for effect)

That seems… perhaps a bit out of scope, maybe not what we were going for with a "root cause". Like, yes, ending capitalism maybe would keep our bar from burning down, but I don't think we have the power to do that in a reasonable amount of time and there's likely to be other solutions available to us that we can implement sooner and more easily.

Detective icons created by Flat Icons - Flaticon
Money bag stickers created by Gohsantosadrive - Flaticon
Creativity stickers created by Stickers - Flaticon

Let's Take a Step Back



We're playing a game I call "trigger whackamole"

Something's missing from our analysis here…

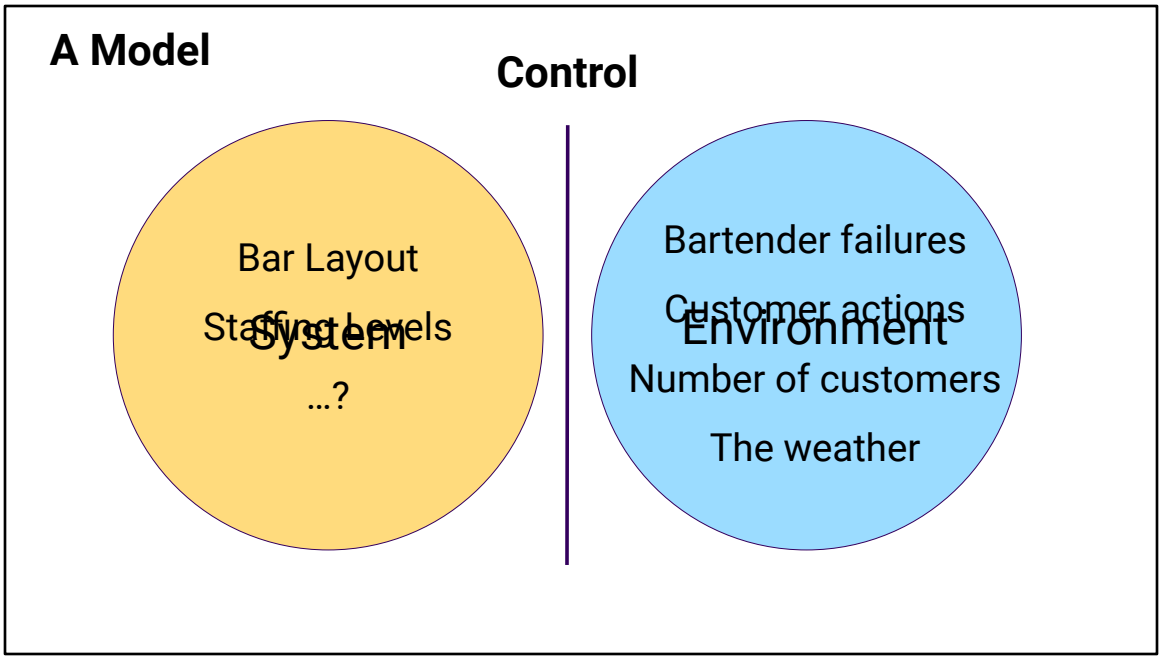
"You didn't use 5 whys right"

- What did we do wrong?
- How should we use it?

So what's actually going wrong?

So something's going wrong here with our incident analysis. I call this thing that we've been doing "trigger whackamole" – instead of finding the actual reliability issues in our system (the real "root causes"), we're finding earlier triggers and trying to resolve those. This doesn't work, because there's always another trigger. And when we do analysis that perhaps finds a non-trigger, it's far too "large" a cause to actually fix (the root cause of every production outage is "the big bang"). (pause)
I'll note at this stage that the wikipedia article on "5 whys" actually has quite a bit of reference to criticisms and shortcomings of the technique (artificial depth, single "root cause", not repeatable). It's also clear that we're doing "something wrong" with our whys. In particular, I think the most important thing going wrong for us is that we don't have any way to judge if our specific "why" is the right one, or if our answer is useful to our analysis. Without a way of understanding clearly what it is we're looking for in our analysis, any method we're using to analyze our outage isn't going to help us.

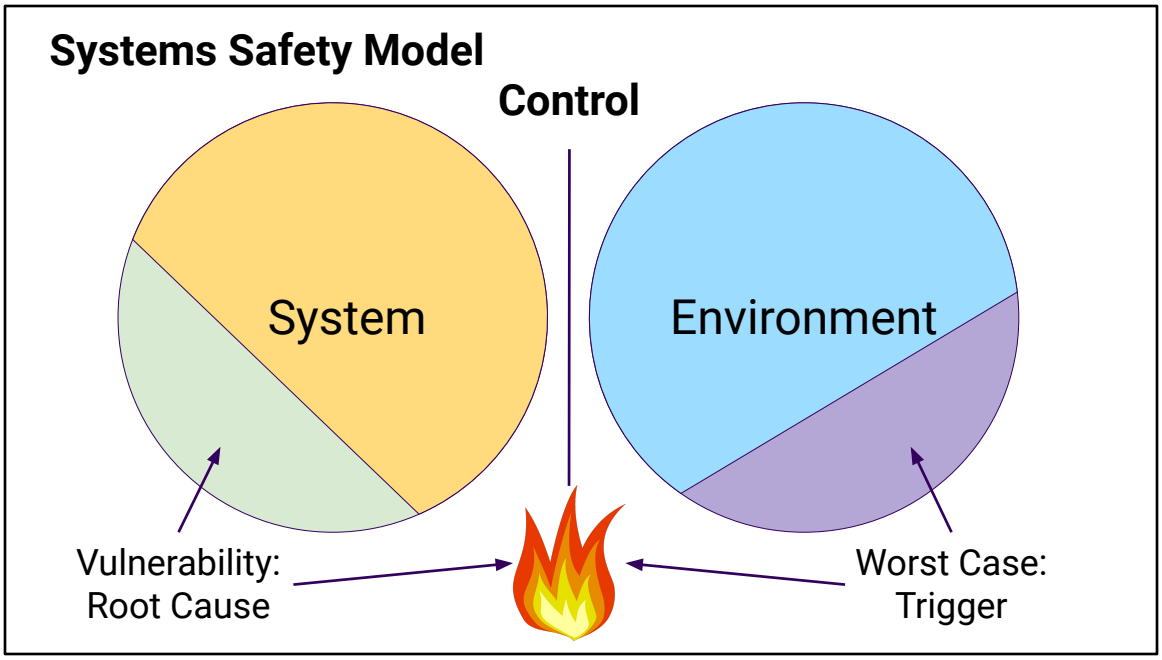
Fun animated icons created by Freepik - Flaticon
Photo by George Becker: https://www.pexels.com/photo/1-1-3-text-on-black-chalkboard-374918/



Let's look at a really simple model of our engineering world, built on principles of systems safety engineering. We can divide the universe into our "system" and our "environment", where the dividing line is "what we can control". <click> For the bar fires we've been looking at, we've been focusing a lot of our analysis on things that are more "environment" than "system" – the things our bartenders do that they're not "supposed" to, actions our customers take, how many customers there are, the weather, … (pause)

We've been *solving* our problems within our system, because that's the only thing we can fix. And in fact one technique we sometimes use when we're trying to fix our system (consciously or not) is to find ways to move some of those environmental factors from the environment into our system – for example, we could have hired a bouncer to keep our bar from ever being too full at once (pause). Classifying system vs. environment is still going to require judgement – but we can be clear with ourselves about which way we've categorized something and why.

What we *haven't* done so far is understood how the thing in the environment, which is our trigger and that we by definition don't control, is actually interacting with our system to result in our outage (next slide)



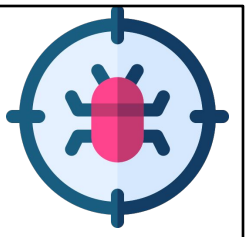
So, here's how we'd think about it in the context of systems safety engineering. We have our system, and in "normal" environmental conditions, it keeps working. But <click> there's some set of environmental conditions that are "worst case" – they're within our intended operating parameters, but they don't happen constantly. They're things like customers asking unusual questions, a sudden influx of traffic, or it being really cold. (pause) At the same time <click>, our system has some ways that it operates where, under those "normal" environmental conditions, it all keeps working just fine. But under those worst case conditions <click>, that vulnerability results in an outage (in our case, a fire). (pause) The goal of any incident analysis should be to find the properties of our system that are interacting with those worst case conditions, and change our system so that it no longer has those vulnerabilities (rather than trying to account for specific triggers). (pause for impact)

The important element is to find the parts of our system that have the *potential* to cause an outage, *in isolation from whether they are being triggered*. This is why it's so important to investigate and remediate near misses – it's the potential for an outage that makes our system unreliable, not the existence of a trigger. One way to imagine it is that our system is full of "outages waiting to happen". (pause) This idea that we control the system and have to design for the environment is a fundamental

part of safety engineering, and a big win for SRE systems analysis. This is also why it's so important to examine our near misses – they can show us when the system has a vulnerability, without the environment being quite bad enough to actually have an outage. (pause)

Fire Clipart from https://pixabay.com/users/clker-free-vector-images-3736/

Terminology Aside

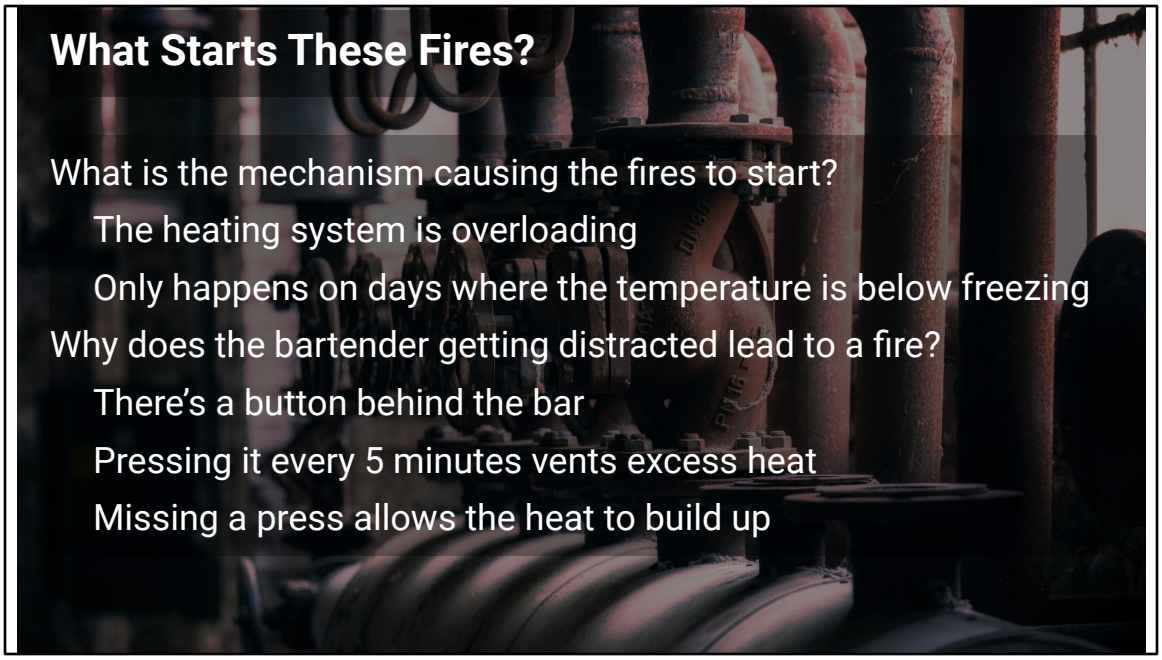


- "Root cause"
  - Yes, this isn't a great term
  - We can agree to mean "the underlying set of vulnerabilities in our system that had the latent potential to cause an outage"

- "Trigger"
  - The set of environmental conditions that "activated" (or could activate) our root cause, resulting in an outage

I want to take a moment to talk about words. I've been saying "root cause" here a bunch, and some of you have probably been very politely not shouting at me about what a bad term that is. I agree! It implies that there's a single factor leading to an incident, and is widely associated with choosing things that aren't systemic vulnerabilities as "the thing to fix". On the other hand, the term is in very widespread use – if you'd like to fight to get everyone to use different language, go ahead. I'd rather "clarify" the definition of "root cause" as jargon that means (read). This also leaves "trigger" meaning (read)

Vulnerability icons created by Freepik - Flaticon
Domino effect icons created by lutfix - Flaticon



So, we need to analyze what's happening *within our system*. With that as our guiding principle, it seems pretty apparent that what we're missing is "what's actually starting these fires?" and "how does bartender distraction lead to a fire starting?". (pause) Notice that we're still asking why – but our "why", and our answers, are focused on our system – how the bar is designed, and what's happening inside it, to lead to the outcome we're seeing, in response to particular triggers. (pause)
Turns out, we have a heating system that overloads and needs to be vented every few minutes, and a button behind the bar that we ask the bartender to press to vent the heater. When they forget to press the button, sometimes heat builds up enough to start a fire. (pause for effect while people map)

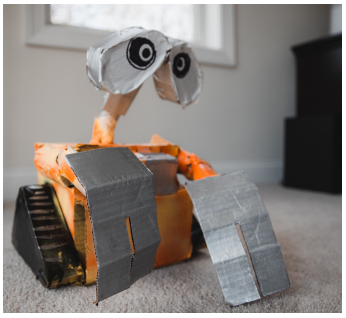
Photo by Peter Herrmann on Unsplash

Solving The Problem

It's natural to focus on the button

Let's press it more often!

…Still relying on a human



Build a robot to press the button

But now we have a robot that can fail

And a forgetful human…

So let's fix our bar! It's pretty natural to focus on pressing that button more reliably. Maybe we'd like to press the button more often, so that forgetting 1-2 presses doesn't start a fire; so a single distraction is less of an issue… but that's still relying on a fallible human to keep things running. We could instead build a button-pressing robot, to automatically press the button every five minutes. And then we've automated ourselves out of a job! (pause). But… now we've made our system more complex. We've added a robot that can *also* fail, and when it does, our human is unlikely to remember about exactly how the robot and the button work, because humans are forgetful like that.

Click here icons created by Freepik - Flaticon
Photo by Edgar Moran on Unsplash

Fixing the System

Why does our heating system overload?

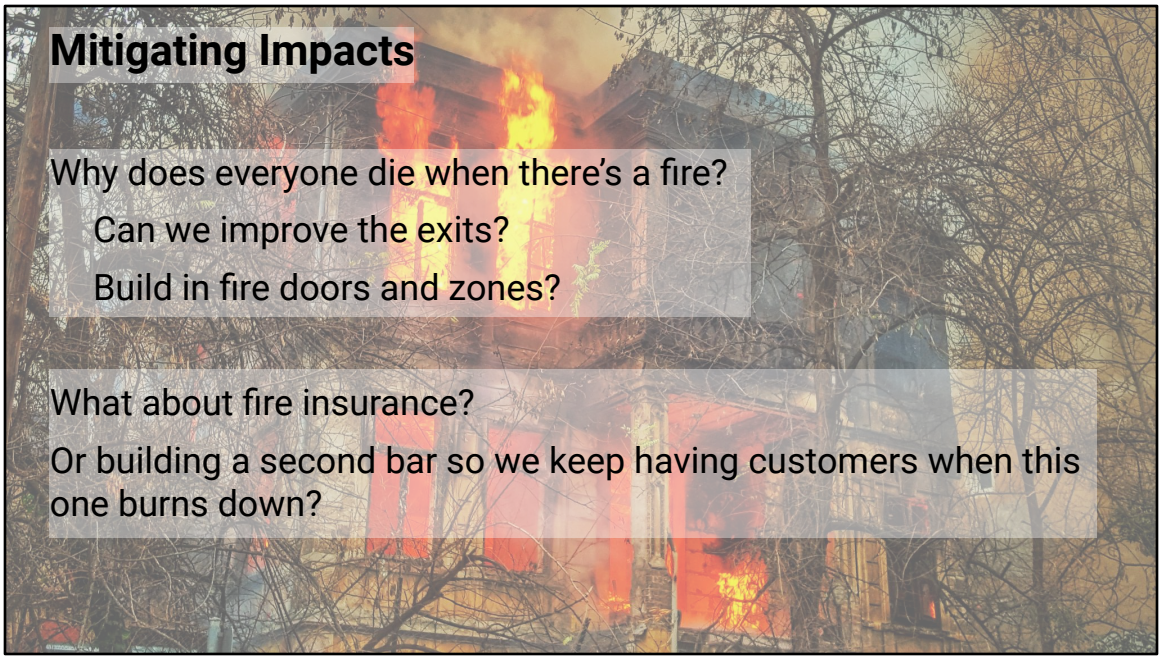
Unreliable? Not enough ventilation? Too high capacity? …

Let's fix that underlying issue

More expensive, but best in the long term

Some problems, the best solution we can build, for whatever reason, is a robot to press the button. But in a lot of cases, we should be asking "do we even need that button?" Can we design our system to not have the button at all?
In this case, we have the button because our heating system overloads. So let's find out why that is, and fix the *design of our heating system* so it's not prone to overloading (or so that it doesn't need manual intervention to keep from doing so, but again, that's less effective). Fixing the overload flaw in our heating is going to be more expensive than just putting a bandaid over the problem, but it will actually make our system more resilient and reliable (short pause). In fact, notice how the button to vent excess was a bandaid over the issue with the heating design in the first place. More components might just give us more ways to fail, if we're not addressing our underlying issues. (pause)

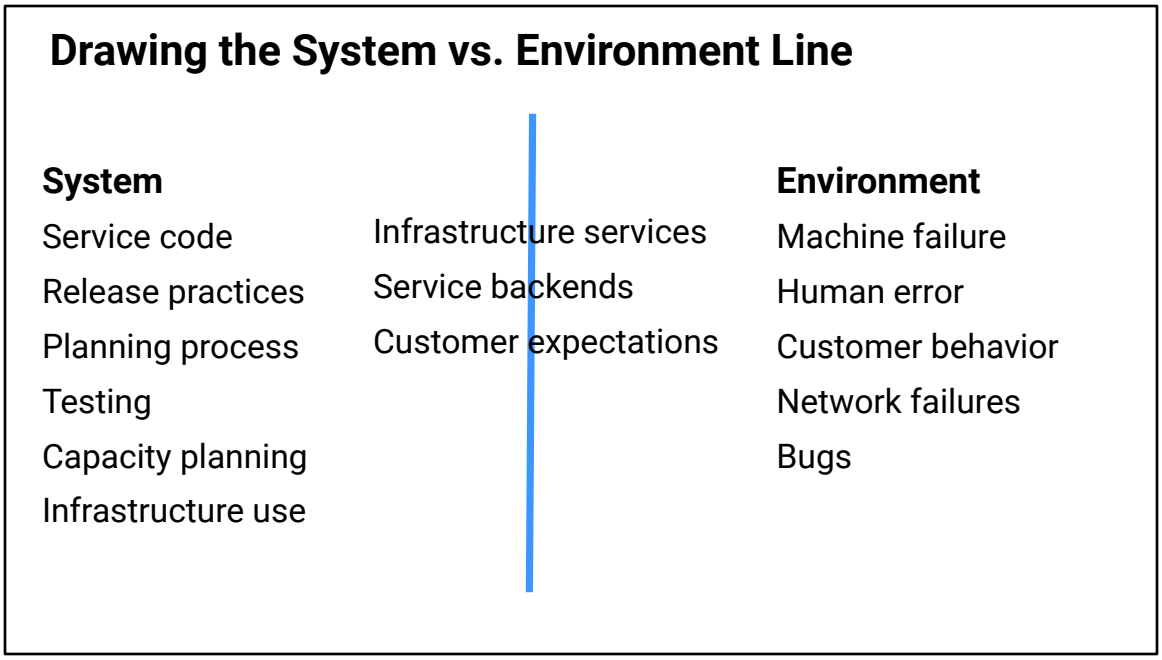
Photo by Felix Mittermeier: https://www.pexels.com/photo/green-tree-photo-1080401/



Another thing we can ask about our system is if, when it burns down, we can have less go wrong – can we keep from having anyone die when the bar burns? Can we make it not matter to our business (very much) when it does? (pause at these examples)

Photo by Chris Karidis on Unsplash

Drawing the System vs. Environment Line



What's in our system vs. our environment can be an engineering judgement call. Some things are "obvious", and some things fall on a borderline (maybe we *could* change them but it involves interacting with other teams and code we don't directly own). What's important to think about is what we can control, vs. what we can't, and have the discussion/do the analysis with that in mind, being explicit on what we think is in scope.

Expanding Our "System"

Designer training?

Design goals?

What about permits?

What's the right scope to solve at?

Finally, we can expand our view of our "system" to look at the social, processes, and cultural elements that lead to problems. In this case, we might ask about the design of this overloading heater. What training did our designer have; what were the cost incentive structures they were designing for (speed vs. labor vs. equipment price vs. maintenance…)? What was the permitting and review process for this design? Notice that as SREs in particular, we start to ask questions about these social elements as we get more senior – the size of system we're trying to fix the vulnerabilities in gets larger.

Photo by NASA on Unsplash


Q&A