Engineering Around ADHD

Erica Ehrhardt

software engineer, <u>Thumbtack</u>

@EricaEEhrhardt • bat-country.us

Hi. My name is Erica Ehrhardt. I'm an engineer at Thumbtack, which is a marketplace to match up local professionals with the people that need their skills. I've been there for four and a half years, formerly on big data infrastructure, now working on our iOS apps. I've been in the tech industry nearly 10 years. I've had an ADHD diagnosis for 13 years, and I've been getting slowly better at dealing with it since then. Hopefully I can cut down that time a little for the rest of y'all in the audience.

Intended audience

- 1. People with ADHD
- 2. Coworkers
- 3. Managers

This talk is for three groups of people. The first is people in the tech industry who find themselves dealing with ADHD: engineers, since that's my own experience, but also designers, PMs, tech writers... we're all in the same industry culture that doesn't always mesh with how our brains work.

The second is coworkers of people with ADHD. I want you to know how we operate and what changes in your workflow might make us more productive.

And the third is managers, especially hiring managers. I might need a new job someday, and now that you know I have ADHD, I have to give you some reasons to not accidentally drop my resume and have it mysteriously land in the trash.

Details I can't cover in 12 minutes

1. Diagnosis and medication:

Talk to an **adult** ADHD specialist

2. Legal protections:

Read up on the ADA

Talk to HR, or a lawyer

There are going to be two major topics I can't cover in this time span. One is the medical side of diagnosis and treatment, the other is the legal protections you may have as a person with a disability. But I encourage you to talk to a specialist... or talk to me after this.

Okay, here comes the fun part. Why should you care about ADHD?

4.4%

of the US adult population

<u>NIMH, 2003, NCS-R study</u>

One good reason is that about 1 in 20 adults has it.

This is a lower bound. The NCS-R study reported American adults with current diagnoses in 2003. For reasons I'm about to cover, the real number might be quite a bit higher.

But what is it, exactly. Attention deficit/hyperactivity disorder. That one dude in middle school who sat behind you, made a lot of noise, and kept throwing pencils at the other kids, right? Actually just the tip of the iceberg.

Researchers started noticing ADHD symptoms in the early 1900s, with standard diagnostic criteria arriving in the late '60s. Unfortunately, research initially focused on certain symptoms and demographics, and public perception followed. Before we break it down into symptoms, let's take a look at that perception.

Symptoms, by the stereotypes

1970s: School-age boys who won't sit still
1980s: School-age boys who won't pay attention
1990s: The adult men those boys grew into
2000s: Wait... girls and women can have this too?
Yes. Yes we can.

Barkley, A Brief History of ADHD

Grade school is one of our screening mechanisms for ADHD. If you don't act out in class, if trouble focusing is your primary issue, well, that's internal. Harder to see. You're likely to be diagnosed much later than you should have been, or not at all. And only recently have we come to understand that it doesn't go away as you grow up.

One study found that **mental health clinicians** miss non-hyperactive ADHD about half the time. So if you're a girl, and you've been pressured very strongly to regulate yourself socially, not make noise, talk out of turn... as far as your **math teacher's** concerned, you can't possibly have ADHD. You're not even going to get a chance for a mental health professional to miss it.

And once you're an adult woman? It's one of the last things anyone looks for.

Symptoms, by the DSM-5 criteria

- 1. Hyperactivity: fidgeting, talking too much...
- 2. Impulsivity: impatience, snap decisions...
- **3. Inattention:** poor time management, procrastination, lack of follow-through, difficulty multitasking, organization & memory issues...

DSM-5 ADHD criteria

Time to break it down. These are the official symptoms of ADHD, the current DSM-5 diagnostic criteria in the US.

Hyperactivity: The most obvious one. Can't sit still. Closely related to Impulsivity: Not **wanting** to sit still. Getting bored easily.

And then there's Inattention. My personal favorite, this is the one that tends to cause the worst problems for adults. Reading the full criteria for inattention is like reading a list of all the reasons my parents, teachers, and managers ever got mad at me. Always late. Can't start things. Can't finish things. Can only do one thing at a time. Only a vague idea of what time it is and where her keys are.

"You just need more self-control."

And in fact, if you have ADHD, you've probably heard this a lot.

ADHD is a neurological issue, not a moral failing.

It's wrong. Let's talk about why.

What are executive functions?

- Hiring
- Firing
- Wearing a suit
- Writing the inevitable
 "Our Incredible Journey" post on Medium when the company gets acquired

Let's talk about executive functions.

Oops. Not those executive functions.

What are executive functions?

- Control of attention
- Switching tasks
- Self-inhibition
- Working memory



And unfortunately, in people with ADHD, these executive functions tend to mature later and to not to work as well as they do in other people.

What do I need executive functions for?

- New sequences of actions
- Overriding habits
- Prioritization
- Troubleshooting
- Technically difficult situations

<u>Norman & Shallice model</u>

Here are some things that humans need executive functions for.

When we can't fall back on memorized behaviors and need to really think.

New situations. Weighing one possible action against another one. Figuring out why something went wrong.

"Technically difficult situations" in the literature probably wasn't meant to refer to the tech industry specifically...

What do I need executive functions for?

- Building new things
- Best practices
- Planning
- Debugging
- Everything we do

... but it should sound familiar, because everyone in this room lives there. In fact, most of the tasks that put heavy demand on your executive functions map directly to stuff we do daily. Working in tech hits your executive functions **hard**.

That sounds bad. How do we deal?

That sounds bad, right?

The biology of ADHD is still mostly an open question.

We know that prefrontal cortex damage can cause very similar issues. But we can't rewire your cortex.

We know that meds like Adderall can reduce symptoms. But they won't reduce them to zero.

So for now, we're stuck with it. How do we deal with it?

Treat it like a tech problem.

We treat it like a tech problem.

Treat it like a tech problem

We've all dealt with tech that's a little... quirky.



In this industry, we've all dealt with tech that's a little... quirky. Let's go with quirky.

It may not perform up to spec, it may be limited in some way, it may have some surprising behaviors. And we're stuck with it.

Treat it like a tech problem

- Buggy scheduler?
- Restricted memory?
- Limited interrupts?
- Expensive context switches?

There are <u>always</u> workarounds.



But we figure out how to get the best out of it anyway, because that's what we do.

So let's throw out the idea that ADHD is a moral failing, and see where we might be able to draw analogies to problems that we already know how to solve.

Treat it like a tech problem

Save your executive functions for where you really need them.

Offload everything you can to:

- Technology
- Rote processes
- Other people



The core concept here is that we want to save our limited executive functions for things we need, like design or debugging, and avoid using them where we can substitute other resources.

And those resources include technology, rote processes that we can memorize, and even other people.

Workarounds for control of attention

Stop the interrupts!

- Turn off popups
- Mute IM channels
- Filter your email
- Put your headphones on



Replace interrupts with polling

Let's look at control of attention. In your office and on your computer, there are a lot of things that want your attention, most of which don't need it right now. But they all steal focus and cause those expensive context switches. And the best way to stop that to make sure you don't see them until you're ready to handle them.

So use your tech to control what you see. Turn off popups and notifications. Mute Slack channels that aren't critical or actionable; at Thumbtack, I asked to move my new team's noisy bots to a new channel to make this easier. Use email filters to separate mailing lists from stuff sent to you specifically. And for in-person noise, especially in the open offices that startups are so fond of, headphones are indispensable.

But how do you not miss **important** messages? Event-driven I/O is the new hotness, but in this case, polling works better. I handle my email and Slack messages mostly when I get in, or right after lunch, already natural breaks in my day. Find yours, make time, and don't worry about interrupts outside of it.

Workarounds for control of attention

One weird trick for staying in the zone:

"Hey, can you take a look at..."

"Sure. Put it in JIRA."

Get it in writing.

And you can enlist your coworkers for help here. When someone has a question, it can be hard not to immediately leap in and help, even if it'd pull you off what you're working on. Trust me, I used to love to do this, until I realized how much time I spent working on other people's stuff instead of mine.

So memorize this phrase: "Put it in JIRA". They already have context that you don't. They can stash that in your team's bug tracker, you can get to it next time you're polling for messages, you don't get distracted from your current task, and in the long term, everybody wins.

Workarounds for working memory

Your bug tracker is your new best friend.

- Can update people so you don't have to
- Store links to design docs, API docs, PRs, other bugs
- Store your work notes
 - Dump your brain here at the end of the day
- Store your plans...

Continuing with that theme, your bug tracker can also be a workaround for working memory. I use ours pretty hard. Anything I can write down and get out of my head is something I don't have to recompute when I get back from a meeting.

It actually doesn't take much time to write down the names of the files I was refactoring, or add a link somebody sent me to the latest design sketch, compared to how much time I'd spend trying to find that info again.

A friend of mine, who's also an engineer with ADHD, got me to start dumping my brain into JIRA at the end of the day. It can be really tempting to spend a little extra time at work wrapping a bug up... and then it's suddenly 9 PM and my cats are wondering where I am. But chances are that bug will still be there tomorrow morning, so instead, I make myself stop working, write down what I was going to do, and **go home**.

Workarounds for working memory

Checklists!

- Separate planning from action
- Don't remember things that your computer can
- Doctors and pilots use them
- Feels good to check items off
- Next step: <u>automation</u>

Gawande, The Checklist Manifesto



And then there's checklists. A low-tech classic. A surgeon named Atul Gawande wrote a book called The Checklist Manifesto, talking about how doctors and nurses, working in one safety-critical industry, copied the idea from pilots and mechanics, working in another safety-critical industry. They use checklists so that they don't forget to close the valve that keeps the plane, or patient, from exploding immediately after takeoff.

As people in the tech industry, we like to think we're smarter than pilots or doctors, so we should absolutely be using checklists too.

At Thumbtack, we use them everywhere. On call for the messaging service? Its playbook has a checklist with exactly the steps I need to diagnose and repair a backed-up message queue. I can go down the list, check check check, not worrying about whether I remembered to run the YAML linter on the new config file, because hey, I already checked that. Once a checklist is stable, it might even grow up and become an automation script.

Workarounds for task switching

Separate planning from action. Literally, if you have to. Work in a different corner. Pick a specific time of day. These are two very different headspaces.



Now those pilots and doctors come up with their checklists ahead of time, when they're not in the OR or on the runway, and are thus in a space more suited to planning than action.

We can use that too. And I'm not talking about full waterfall here. When I was at Microsoft working on Office, we sometimes had a three-year space between planning and action. Bit much. I'm talking about the tactical level: how do I solve this bug, how do I ship this feature.

I'm at a desk all day, like most of us, and I've found that my desk is not a great place to plan. I tend to get stuck in the coding mindset: sit down, open Xcode. However, if I grab my laptop and go find the furthest corner of the office, I get a lot more done. Something about the change in physical environment helps convince my brain to do something else.



Finally, here's a strategy anyone in distributed systems will know: throw more cores at it. Pair programming, 1:1s, talking through a design over coffee, these are all ways you can ask another person to help you start something you can't manage to start yourself.

In my own experience, just a few minutes bouncing stuff off another person on my team can save me a few hours of being stalled out and not sure how to continue, or let me figure out where to cut something that got a little overgrown.

Are there **any** upsides?

Okay, we've been talking about workarounds for a while now. Maybe a little bit grim, admitting that I have limitations that require workarounds.

Can we end on a happy note? Are there any upsides to ADHD?

Sort of.

Sort of.

Are there any upsides?

- Creativity: The flip side of impulsivity
- Different filters
 - You'll see annoyances that other people filter out: repetitive operations, UX hiccups, slow performance
- Hyperfocus
 - Can exceed neurotypical performance on inherently interesting tasks, or under high stress

<u>Glickman & Dodd</u>

Impulsivity, for example, sometimes means that we will try ideas that others miss. We may also see subtle annoyances that other people just filter out as part of the status quo for your product.

And the flip side of glitchy control of attention is a state called hyperfocus. There are studies showing that, on certain tasks or under high-stress conditions, people with ADHD can actually far outperform people without it. You've heard of flow, or getting in the zone? This is their big sister. You don't want to rely on it all the time, but ... a lot of people with ADHD have at least one story about the time they successfully refactored an entire codebase in 18 continuous hours.

Thank you.

So that's my talk. Thank you for coming, and I hope this helps.