EPISODE 837

[INTRODUCTION]

[00:00:00] JM: CamelCamelCamel is a tool for tracking prices on items on Amazon.com. The company was launched 11 years ago and is built off of the Amazon product advertising API. Daniel Green Green is one of the founders of CamelCamelCamel, and he joins the show to describe his experience building the product. I'm going to refer to CamelCamelCamel as C3 for the rest of this preamble.

Amazon and C3 have a complex relationship. C3 makes its money from referral listings. When a user tracks the price of an Amazon item using C3, that user will probably eventually click on the referral listing on C3. If a user purchases the item on Amazon using that referral link, C3 will get a percentage of that purchase from Amazon.

C3 has millions of product listings where they are tracking the price of items on Amazon. They've created a directory with a large subset of Amazon's items by leveraging an API that was originally meant for advertising. Whether or not this is a proper use of the API, C3 arguably leads to more purchasing volume on Amazon, which is ostensibly why Amazon lets the company continue to operate.

In today's show, Daniel gives a history of CamelCamelCamel, including his own background as an engineer working in finance and how he wound up working on C3 for the past decade. We also talk about how C3 is architected.

Daniel is a fascinating character, and I had a great time talking to him. I found him to be extremely honest. To be a quite different thinker than a lot of the people I've had on the show. You'll see what I'm talking about as you'll listen to this episode. It's quite a good one. So thanks for listening.

The FindCollabs Open is our second hackathon for FindCollabs. FindCollabs is the company I started. It's a place to find collaborators and build projects. If you create a project before June 15th, 2019, you will be entered into the hackathon. We've got prizes for the best Flutter project, the best machine learning project. Music, if you want to write a song, game design, data visualization. We've got all kinds of cool projects and you can find out about how to enter your project by going to findcollabs.com/open. It's not that complicated, and it's a great place to find some collaborators and build some sweet products, whether you're building software or something else.

Thanks for listening and let's get on to today's show.

[SPONSOR MESSAGE]

[00:02:50] JM: DigitalOcean is a reliable, easy to use cloud provider. I've used DigitalOcean for years whenever I want to get an application off the ground quickly, and I've always loved the focus on user experience, the great documentation and the simple user interface. More and more people are finding out about DigitalOcean and realizing that DigitalOcean is perfect for their application workloads.

This year, DigitalOcean is making that even easier with new node types. A $15 flexible droplet that can mix and match different configurations of CPU and RAM to get the perfect amount of resources for your application. There are also CPU optimized droplets, perfect for highly active frontend servers or CICD workloads, and running on the cloud can get expensive, which is why DigitalOcean makes it easy to choose the right size instance. The prices on standard instances have gone down too. You can check out all their new deals by going to do.co/sedaily, and as a bonus to our listeners, you will get $100 in credit to use over 60 days. That's a lot of money to experiment with. You can make a hundred dollars go pretty far on DigitalOcean. You can use the credit for hosting, or infrastructure, and that includes load balancers, object storage. DigitalOcean Spaces is a great new product that provides object storage, of course, computation.

Get your free $100 credit at do.co/sedaily, and thanks to DigitalOcean for being a sponsor. The cofounder of DigitalOcean, Moisey Uretsky, was one of the first people I interviewed, and his interview was really inspirational for me. So I've always thought of DigitalOcean as a pretty inspirational company. So thank you, DigitalOcean.

[INTERVIEW]

[00:04:57] JM: Daniel Green, you are a founder of CamelCamelCamel. Welcome to Software Engineering Daily.

[00:05:03] DG: Thank you for having me.

[00:05:04] JM: What is CamelCamelCamel?

[00:05:09] DG: That's our website. It's an Amazon price tracking service where you can tell us what price you'd like to pay for individual products and we'll email you and the price drops to that level. You can also look at the historical prices for individual products. So you can decide if the current price is good or not relative to its past.

[00:05:33] JM: Why do people use a third party price tracker? Is this feature not available directly in the Amazon marketplace?

[00:05:43] DG: Well, usually, I think people will add something to their shopping cart and just wait for that little notification that says the price has changed, but you have to log in to your Amazon shopping cart to see that message, whereas we will send you an email when that happens.

[00:06:02] JM: Explain the business model of CamelCamelCamel.

[00:06:06] DG: Well, for a long time it was simply an affiliate marketing thing where we get a small percentage of every sale that we refer to Amazon. More recently, we've gotten into display advertising also.

[00:06:22] JM: But the affiliate marketing thing, that still does work?

[00:06:25] DG: It does.

[00:06:26] JM: Interesting. Explain the name; CamelCamelCamel.

[00:06:32] DG: Oh, I wish I had a really great story for you, but it was just a domain name that I had before I came up with the concept of the website. At the time, I was just writing code like crazy, developing any idea that I had just as – That's what I did for work and what I did for leisure at that time. So I found that you could get Amazon pricing data easily and directly from Amazon. I just started to build out a little demo of what I could do with that data and eventually needed a place to put it online where I could share it with my friends, and that was the domain that I chose.

[00:07:15] JM: What are your sources of Amazon data?

[00:07:20] DG: Amazon's affiliate program has an API that lets us query product data.

[00:07:26] JM: How rich is that API? What's an API request or what are the different requests you can make?

[00:07:34] DG: It's gotten leaner overtime. You used to be able to get access to a user's wish list contents, for example, and that part of the API got deprecated awhile back. You used to be able to get, say, a list of all of the merchants' offers for a specific product. Now it's really only the lowest price offer that's available, which for our purposes are still usually plenty.

[00:08:07] JM: How does Amazon feel about CamelCamelCamel? I could see them kind of going in either direction being supportive or being a bit affronted.

[00:08:19] DG: Yeah. We've heard it's about 50-50 internally. Some people think we are adding sales. Some people think we're just deferring sales that would have already happened.

[00:08:37] JM: What do you think?

[00:08:40] DG: I think people are always going to look for deals and pretty much everybody wants to save money. So I feel like we're adding some value there. Certainly, if you're planning to buy something and you see the price history is not great. Maybe you're going to delay that, but maybe if you're more price sensitive the most, you would just never buy if you think it's not going to come to a good price and you see our alert and then you buy it.

[00:09:13] JM: Describe your engineering stack.

[00:09:18] DG: We are primarily a Ruby on Rails operation, starting with Rails 2 and upgraded to 3. Currently playing around with upgrading 4, just because we're kind of at far past the end of life there. Our price history charts are rendered in PHP 7 however.

[00:09:44] JM: Tell me more, like databases, data infrastructure. Give me a more gratuitous lay of the land.

[00:09:52] DG: Sure. So a request comes in. It hits nginx and HAProxy. The reason we use both of those is kind of just a historical thing. Back when HAProxy was a little bit less fully featured, we needed nginx to be able to send specific requests to specific servers. So if a request comes in to our product search, which ends up making an API request to Amazon and, thus, taking a little bit longer to return to the user, we send that to a handful of servers that only handle search requests so that, say, request to the homepage don't get stuck waiting for those.

So requests come in through there, they hit our webservers, which are running Apache and Phusion Passenger getting processed by Ruby of PHP. Database is MySQL. Recently made the upgrade to 5.7, and we're running the Percona flavor of that. For caching we use Memcached.

[00:11:04] JM: Let's talk about the core the process that you have to do. So in order to build these price models for CamelCamelCamel, there is a repeated usage of this Amazon advertising API that gives you some data on the price. Tell me how that scraping infrastructure works.

[00:11:27] DG: Sure. So essentially what we do is build a queue of – Or multiple queues of products. We split up things in different ways by the Amazon country. Because, of course, we support all of Europe and North America. Well, Canada and the United States. So we split that up.

We also prioritize based on user interest. Since we have a finite number of API requests, we have to try to make the most of those. So whether a product is being actively tracked by a user or not, it gets higher priority. Then we use Amazon SQS to create these queues and then we just pop things off the queues and make API requests.

[00:12:21] JM: So just to be clear. You basically build queues full of request for every product in Amazon.

[00:12:26] DG: I wouldn't claim full coverage, but it's over 100 million products at this point.

[00:12:34] JM: So you got 100 million products. How many products in the United States you know off the top of your head?

[00:12:41] DG: I don't. I would guess more than half.

[00:12:44] JM: Okay. So let's say there're 50 million products in the U.S. catalogue that you cover the price for. So you're saying that basically you have like – Is it like a round robin queue where you're just like constantly making requests over those 50 million items or how does that work?

[00:13:05] DG: Yeah. We're essentially just – We build the queue. We have a number of workers that pull items off of it, make the API request. Then when the queue is empty, we rebuild it, and that's running 24 hours a day.

[00:13:22] JM: How do you find new products to add to your round robin queue for a given geo?

[00:13:30] DG: A couple of ways. We have our product search which makes an API request to Amazon. Basically, the two main API requests that we make are for product information and for search results. We figured early on that the best search results were probably going to come from Amazon themselves. So we just pass that on to them.

Anything that we see in those results that is not in our database, we save and start and tracking. Then the other way is our browser extensions. If a user activates those on a product page for a product we're not aware of, we will add that to our list also.

[00:14:14] JM: How well documented are these APIs for like searching Amazon or loading Amazon information?

[00:14:24] DG: It's really just like any other AWS API.

[00:14:30] JM: Oh really? Okay. So they make their APIs like pretty accessible to external users.

[00:14:39] DG: Somewhat, less and less. Since it's this API in particular is for affiliates, they have certain restrictions. Of course, you need an active affiliate account. The number of requests you can make in a given time period are based on your affiliate sales, that kind of thing.

[00:15:01] JM: Have you ever run up against rate limiting?

[00:15:05] DG: Sure.

[00:15:07] JM: So when that happened, did you have to just like tone down the number of requests or did you find some way around it?

[00:15:14] DG: Both. I mean, it kind of depends on the cost. So given that we have a finite number of requests, our system is generally tuned for the 10 months of the year that are not the holiday shopping season. So we try to maintain a certain update frequency for products during the year.

Then once we get to the holidays, product search becomes much more important, so we kind of shift resources over to that. But early on, it was really more about getting our sales to a certain point and using the API as efficiently as possible. So moving away from our original code, which just requested one product per API call, realizing you could make up to 10, say, and then, “Oh! You can actually double up API requests.” So now you can update 20 at a time.

[00:16:20] JM: Wow! Was that like just random discovery through kind of hacking on it and reverse engineering what was going on?

[00:16:32] DG: Some of it, say, realizing you can update a request information for 10 products at once was just reading the documentation. Who wants to do that?

[00:16:42] JM: Yeah.

[00:16:44] DG: Other things were digging deep into like the AWS forums and trying to learn from other people’s experience there.

[00:16:59] JM: Really? So there’s like a community of people that are doing stuff. I don’t understand, what does it have to do with AWS? I assumed this is like the Amazon advertising API. I thought that’s the main API you’re using. So that’s under the AWS brand of things?

[00:17:19] DG: Maybe not officially. I mean, that’s where the forum lives. The API responses seem to indicate it’s all kind of AWS in the backend. But you’re not going to find the product advertising API listed if you log in to the AWS console. I think it’s maybe just sharing a backend.

[00:17:43] JM: Got it. So who is in this forum community of people that are working on stuff like this? Like not specific names. I’m just like wondering what is the characteristics of the kinds of platforms people have built on these kinds of APIs.

[00:18:00] DG: Well, I think you have a lot of people who are just building affiliate stores, really novice users who are looking for help doing the absolute minimum basic things. You have people who are discussing how to bring the most out of the API, which is more like what we’re after. People just looking for like tech support from Amazon, and Amazon does respond on there occasionally.

[00:18:38] JM: Give me a little bit more context on that holiday peak season stuff. So you said product search becomes more important. Is that to say that you have to get more breadth and perhaps less sample frequency in the holiday times?

[00:19:00] DG: Well, I would say it's really just the result of the additional website traffic that happens. So we've got products that no one is tracking, but we like to keep updated just in case somebody starts tracking them. Then we've got the products that we really consider a high-priority, we have a potential sale on. So we want those to stay really updated.

During the holidays, we get quite a big spike in traffic. So that's more search requests that are taking API requests away from updating products. So we kind of have to lower the rate at which we update the lower priority products to leave room for search.

But in some cases, if, say, the Today's Show mentions us, sometimes we just run out of API requests. You get 30,000 people hitting your site at once and there's not a lot we can do but hope they're all not searching for unique keywords.

[00:20:16] JM: Are there people who their shopping habits on Amazon are like primarily through the lens of CamelCamelCamel?

[00:20:31] DG: It's possible. I mean, we don't do a lot of invasion of privacy style analytics. So it would be hard to say. But there's definitely people with thousands of items in their watch list. So I imagine they're doing a fair amount of shopping through us.

[00:20:56] JM: What other kinds of conversations have you had with Amazon, the company?

[00:21:05] DG: It primarily boils down to how's it going, what are you guys up to? Are you still being good affiliates, that kind of a thing. But we can't really discuss too much of our relationship with Amazon.

[SPONSOR MESSAGE]

[00:21:35] JM: When a rider calls a car using a ridesharing service, there are hundreds of backend services involved in fulfilling that request. Distributed tracing allows the developers at the ridesharing company to see how requests travel through all the stages of the network. From the frontend layer, to the application middleware, to the backend core data services, distributed tracing can be used to understand how long a complex request is taking at each of these stages so the developers can debug their complex application and improve performance issues.

LightStep is a company built around distributed tracing and modern observability. LightStep answers questions and diagnosis anomalies in mobile applications, monoliths and microservices. At lightstep.com/sedaily, you can get started with LightStep tracing and get a free t-shirt. This comfortable, well-fitting t-shirt says, “Distributed tracing is fun,” which is a quote that you may find yourself saying once you are improving the latency of your multi-service requests.

LightStep allows you to analyze every transaction that your users engage in. You can measure performance where it matters and you can find the root cause of your problems. LightStep was founded by Ben Sigleman, who is a previous guest on Software Engineering Daily. In that show he talked about his early development of distributed tracing at Google. I recommend going back and giving that episode a listen if you haven’t heard it. If you want to try distributed tracing for free, you can use LightStep and get a free t-shirt. Go to lightstep.com/sedaily.

Companies such as Lyft, Twilio and GitHub all use LightStep to observe their systems and improve their product quality.

Thanks to LightStep for being a sponsor of Software Engineering Daily, and you can support the show by going to lightstep.com/sedaily.

[INTERVIEW CONTINUED]

[00:23:46] JM: You said that you have to acquire a certain volume of affiliateship in order to maintain in kind of the good graces of the API rate limiters. How has that ratio fluctuated overtime?

[00:24:06] DG: I should say, first of all, when we started, it was more open-ended. So they didn't have this sales-based requirement. But overtime, I think they discovered that this API was just kind of a gaping hole in their data security, where people were not using it for affiliate request at all and were just using it as a data source, say, to sell the data or to use as research for unrelated things. But by the time those quotas came around, we were already maxing them out. So it hasn't been much of a concern for us.

[00:24:55] JM: I imagine the volume that you're giving them through affiliate links. I mean, that's got to be significant enough to qualify for whatever is the maximum API availability, right?

[00:25:10] DG: Yup. Yup.

[00:25:12] JM: Have you ever asked them for like special treatment, like additional API access, or is the relationship not there?

[00:25:22] DG: Well, I think once you reach a certain volume, you end up getting a shoulder tap from an account manager, and that's what happened. But we can't really discuss the specifics of our relationship beyond that point.

[00:25:40] JM: Fair enough. What is it like more generally building this kind of platform off of – I mean, it's like APIs that are available, but it's not like exactly used in the way that they were intended originally, but like you found this way that your relationship with Amazon is – I mean, like you said, it's like 50-50 depending on which Amazon employee you would talk to. Again, I could see it going either way. But just experientially, like as an engineer, what is it like building something in that sort of nebulous territory?

[00:26:20] DG: Well, this is definitely just my opinion. Obviously, I'm not inside Amazon. But it seemed to us that early on, we just kind of went unnoticed. Then once our site became popular enough that it warranted that kind of shoulder tap from an account manager, it seemed like the affiliate and API rules started changing sort of to very narrowly allow what we were doing.

But as a lot of people have said over the years, building a business that's entirely related or dependent on someone else's is risky at best. We certainly live in kind of a constant state of stress not knowing if opinions could change or whatever inside of Amazon to where they could shut down our business if they wanted.

[00:27:24] JM: Have you thought of kind of like backup strategies or backup businesses that you could go down the path of?

[00:27:34] DG: We are always kicking ideas around trying to find something that's interesting for us to work on. In the last year, we launched a site called Camel Lodge, which is a hotel price tracker. So not entirely learning from our potential mistakes there, but at least having two different sources of stress.

[00:27:58] JM: Actually, that's an interesting point. I mean, I wonder – So you must have asked yourself. What are the other things that we might be able to creatively scrape and use our core competency?

[00:28:09] DG: Yeah.

[00:28:10] JM: I mean, I guess it amplifies – But the surface area. I mean, because you have to keep up with these different changing APIs. Then, yeah, I can see how this is just ever shifting mound of quicksand.

[00:28:23] DG: Well, thankfully, an API, especially in Amazon's case, they're fairly thoughtful about API changes and how they affect their users. It's much more reliable than trying to scrape, which is something we've always avoided. It's just we're not interested in playing that cat and mouse game. Amazon's got more resources to prevent scraping. So we're just not going to do it. I think that probably applies to anyone who'd be interested in price tracking. Hotels.com or whoever is a massive business, they're well ahead of us in terms of preventing scraping.

[00:29:06] JM: Right. There are like some third-party data brokers that do like scrapy stuff, and you could like kind of reduce your reliability if you wanted to, or you could go through these data brokers. But I guess that reduces the quality of data, and maybe that's just in disjunction with your ethics.

[00:29:30] DG: It definitely would make us feel like bad affiliates in Amazon’s case, and based on what I’ve seen, would also be tremendously expensive, especially for the kind of volume that we do in terms of API requests.

[00:29:51] JM: Speaking as a total like bystander, how would you evaluate the quality in those data broker kind of things, where they are scraping Amazon and reselling data. Is it junk or can people actually do it with good data?

[00:30:11] DG: It’s a good question. I think it gets – If you wanted to verify, you just be like spot checking, right? Making API requests through the scraper and look with your eyes, but it ends up being kind of shady in our view.

[00:30:30] JM: Yeah, to even like touch it with [inaudible 00:30:31].

[00:30:32] DG: Yeah, because they're probably using like residential proxies and who knows how they've acquired those. Of course, they’re breaking Amazon's terms of service, scraping on the status. So this is not something we’re interested in.

[00:30:50] JM: So used to work in finance, right? You used to do like trading related things I assume?

[00:30:56] DG: Yes.

[00:30:58] JM: How does that experience mapped to what you're doing with CamelCamelCamel?

[00:31:05] DG: Well, I think it kind of – In some ways it made us realize maybe we were not necessarily helping the wrong people, but there are people who need help more in terms of making or saving money. It's great if you're helping a hedge fund who’s helping people's retirement accounts grow. But it's a much more direct relationship to have to help consumers. It feels great when they email you and say, “Hey! Thanks. You save me on this Christmas gift for my kids.”

[00:31:49] JM: Yeah. What's striking – My first job out of school was at a trading place, and that was the conclusion that I came to. It's also the conclusion that a couple of other people I've interviewed on the show came to, where it's like working in the trading world is very interesting. The people are insanely smart, but it's kind of like building a better mousetrap. You look around – When you're working with traders or working for a hedge fund or whatever, you kind of look around – If you look beyond the finance industry and you see, "Oh! I'm really just building a better mousetrap," and these people out there, they're actually building new mousetraps.

[00:32:34] DG: Yeah. I mean, even the personality differences between my team and the people we used to work with, it just wasn't us.

[]00:32:49JM: What are those personality differences? Because I agree with you, in finance you have these people who are razor-sharp intelligent, super creative. Oftentimes, they're like a classical pianist on the weekends, but there is some cultural difference between the finance tech world, meaning trading tech, and that of like – I don't know what you would describe yourself, as may be like a builder or entrepreneur or something. What is that cultural difference?

[00:33:18] DG: The first thing that comes to mind is like a type A personality, right? To me, there's – Of course everyone that we work with was super smart, like you said, and I would apply that to my coworkers also. But we're not salesman. I think to some degree, everybody in that business has that kind of type A personality where they're super competitive, they're really driven, which is not a bad thing. But I don't know. Something about it just was kind of abrasive to us.

[00:34:00] JM: I have used CamelCamelCamel the most on one particular item. It is called Dr. Elsey's Ultra Premium Clumping Cat Litter, 40 Pound Bag. I don't know why this thing fluctuates in price so much, and like maybe because it's heavy or something. What was interesting about the Dr. Elsey's Ultra Premium Clumping Cat Litter, 40 Pound Gag is it's one of the rare items where I have seen Amazon basics try to clone it and fail. Amazon basics cat litter came out. It wasn't great. Then they discontinued it. It was really interesting just following this. You can follow price trends, and I end up buying on Walmart sometimes, because the price is actually different on Walmart sometimes. Do you have a perspective for what causes the most price flux in Amazon prices?

[00:34:55] DG: It's kind of a black box. I mean, you have Amazon itself and then you have the third-party marketplace, which is an enormous crazy competitive thing where you could watch two ROBO pricers compete for the lowest offer, or maybe the highest. So you could have a textbook that becomes priced at $1 million because somebody forgot to set an upper limit, and these two bots just go back-and-forth pricing.

But on the Amazon side, we've seen some really interesting trends not really based on anything that we could notice. Just it seems like Amazon is constantly experimenting with prices, changing them just a little bit, stepping them down, stepping them down. Then once there's a purchase, shooting back up and starting over again.

[00:35:59] JM: So you're seeing Jeff Bezos' roots as a hedge fund person himself, showing their teeth perhaps.

[00:36:08] DG: I think they're definitely experimenting.

[00:36:14] JM: Did you see any new trends when Alexa came out?

[00:36:18] DG: Not that I recall.

[00:36:21] JM: So one thing that happened to me recently, and I've heard this happens, is I use my Alexa to order some Swiffer Dry Pads, and sometimes I go aggro on things and I'm like, "Yeah, just buy it outright." Other times I'll add it to the shopping cart and I'll see what item gets added to my shopping cart.

In this case, what got – I went aggro with it when I hadn't ordered it. Unbeknownst to me, a $107 purchase for somebody who had SEO'ed their way to the top of the Alexa requests for Swiffer Dry Pads was able to take my money, and one complaint to the customer service later and this third-party seller was giving me a free refund and just asked me to donate the Swiffer and paper towels to charity of course so they don't get caught by Amazon. But it's kind of hilarious, like seeing the different ways that people kind of backdoor into little shady tactics. Have you seen any like a weird shady tactics and fraud schemes across Amazon?

[00:37:26] DG: Yeah. I mean, both. Regarding fraud, there's always the contact us before making a purchase people, which so –

[00:37:37] JM: What does that mean?

[00:37:37] DG: So every seller has a little blurb they can put in when they offer a product. Essentially, they're trying to take the transaction offline so they can rip you off, or off of Amazon rather.

On the other hand, it's not technically fraud perhaps, but we are aware of a lot of merchant training seminars and things coming out to places like China where they teach you how to work your way up to the top of the first page of the search results on Amazon, or what the best way to sell your knockoff products are, things like that.

[00:38:22] JM: My limited dablings into reading about like the Amazon third-party marketplace, or kind of these weird nested LLCs and stuff that set up really, really weird, perhaps, money laundering, but it's impossible to tell for whatever Bloomberg reporter or somebody was looking into this. But this ecosystem is just gigantic. Oftentimes, people who order on Amazon, who just order on Amazon, have no idea of the scope of the ecosystem. Do you have any anecdotes that kind of illustrate just how tremendously gigantic the Amazon third-party ecosystem is?

[00:39:06] DG: Will, I think just the fact that people have scammed it to such a high degree is proof of that. There was maybe like a year or so that people were blaming us because they were getting scammed on Amazon by the people who want you to take the transaction off Amazon, and they're just going to steal your money.

The problem there is that the API doesn't really give us much merchant information. So there wasn't any way for us to tell, "Oh! These guys are obviously scammers." But eventually it seems like Amazon started paying more attention to that.

You also have – I forget when it was, maybe like 2014. I want to say, Warner Bros. sued Amazon, because their marketplace was full of counterfeit DVDs. Apparently, the way the

Amazon fulfilled by Amazon program works is they just throw everything into a pile and intermingle the inventory. So once a counterfeiter gets their product in, who knows? You could be ordering from one merchant and getting someone else's counterfeit DVD.

[00:40:33] JM: When you see the scope of Amazon's marketplace and the scope of AWS and you have experience in the financial industry, you know what can go wrong. Do you ever worry? And this is like totally outside of your business scope, more as a human being. Do you worry that Amazon is too big to fail?

[00:40:59] DG: I certainly worry the I won't be able to watch Netflix if AWS goes down. I got to get my West Wing on.

[00:41:09] JM: No, seriously. I'm being completely serious.

[00:41:12] DG: Sure. I mean, that's just one example. I mean, Netflix is a publicly traded company, I believe. That I could have stock market implications if their business takes a big hit. Of course Amazon is a massive public company. So if AWS has problems, that's going to affect not only Amazon.com, but everyone who relies on it.

[00:41:37] JM: Right, and that's my point. It's government infrastructure, probably the power grid. I just think it's interesting, because it's like this – Well, whatever. I mean, I guess you have no choice but have some single points of failure in every architecture.

[00:41:52] DG: I suppose. I hope that the trust people place in Amazon's cloud is not misplaced given that they have many data centers. It seems fairly resilient, but you still see whole services going down. It's like how do you avoid that? I don't know if breaking up Amazon into pieces would really fix that.

[00:42:23] JM: No. No. I don't think so. I don't think so. I think it's just interesting. It's an interesting characteristic of our times.

You do seem to have an aversion to centralization or something, or maybe it's data privacy infringement. For example, in our interactions, you did not have a LinkedIn. Can you tell me more about like what are your beliefs around what our privacy should be on the internet?

[00:42:52] DG: A lot of it I think comes to what you choose to give up, and whether or not you understand what you're giving up. Say, you sign up for Facebook and you upload your photos or anywhere. The terms of service are a mile-long. Nobody reads them, but they now have rights to your photos potentially, or the fact that you can talk about something with your friend and then the next time you're scrolling Instagram you're getting ads for that. I don't think anybody would give permissions if they were asked about it, but they are installing Instagram. So that's just something I choose not to do knowing not trusting anyone.

[SPONSOR MESSAGE]

[00:43:57] JM: Triplebyte fast-tracks your path to a great new career. Take the Triplebyte quiz and interview and then skip straight to final interview opportunities with over 450 top tech companies, such as Dropbox, Asana and Reddit. After you're in the Triplebyte system, you stay there, saving you tons of time and energy.

We ran an experiment earlier this year and Software Engineering Daily listeners who have taken the test are three times more likely to be in their top bracket of quiz scores. So take the quiz yourself anytime even just for fun at triplebyte.com/sedaily. It's free for engineers, and as you make it through the process, Triplebyte will even cover the cost of your flights and hotels for final interviews at the hiring companies. That's pretty sweet.

Triplebyte helps engineers identify high-growth opportunities, get a foot in the door and negotiate multiple offers. I recommend checking out triplebyte.com/sedaily, because going through the hiring process is really painful and really time-consuming. So Triplebyte saves you a lot of time. I'm a big fan of what they're doing over there and they're also doing a lot of research. You can check out the Triplebyte blog. You can check out some of the episodes we've done with Triplebyte founders. It's just a fascinating company and I think they're doing something that’s really useful to engineers. So check out Triplebyte. That's T-R-I-P-L-E-B-Y-T-E.com/sedaily. Triplebyte. Byte as in 8 bits.

Thanks to Triplebyte, and check it out.

[INTERVIEW CONTINUED]

[00:45:47] JM: Let’s say Instagram explicitly said, “You can either $5 a month for the service for no ads and no surveillance, or you can get surveillance plus ads.” So you think people would choose the $5 or do you think they would opt out of the service completely and so they can avoid making that choice, or do you think they would choose surveillance?

[00:46:07] DG: Probably some of both. I think a lot of people would probably bail if it was a choice between paying or knowing the full depth of the surveillance.

[]00:46:23 JM: Really? Because I feel like people – Many people have intuited, like I have intuited at this point. I'm just kind of like, “Whatever.” It’s just kind of like a new paradigm, right? We don't really have privacy, I mean, to a large extent. I mean, there are areas of the world that we still can carve out privacy. For example, my podcast listening habits, the ad tech around that is still pretty immature. So like I can listen to a podcast on something but will not be recommended ads. It’s only a matter of time, to some extent, but I don't know. I mean, is that cynical or is that like – Am I progressive or am I wrong?

[00:47:01] DG: I think it's up to the individual user. It’s not something I'm interested in. I definitely seem to be in the minority. Everybody just wants all the free new cool stuff. Of course, there's value in being able to share photos with your friends and stuff. But it doesn't seem worth it to me for the constant surveillance and advertising.

[00:47:31] JM: Are there any conveniences in your life that you have adapted despite their exposure to the surveillance state.

[00:47:42] DG: I begrudgingly have a home pod in the house. I own an iPhone, whether or not that's very deep surveillance. Apple certainly claims they’re big on security. Do I necessarily trust them 100%? But I try to limit it is much as possible.

[00:48:08] JM: I heard a pretty interesting podcast recently where they were talking about the fact that like every phone and computer these days has a microphone on it. You can't opt out of it seemingly. I mean, this is getting real speculative. We'll get back to CamelCamelCamel on a sec, but I'm kind of intrigued by your thought process, because you're a little bit different than a lot of people I've talked to.

Could you imagine a world where like there's a product or like a set of products that are just like premium products in regard to how secure their supply chain is and how devoid of like microphones and other kind of surveillance technology are on them?

[00:48:48] DG: I wouldn't be surprised if that's something that appears, especially as you have so many high-profile people and smart tech people who probably would want that. I assume Elon Musk would trade his smartphone for something that was more secure if he could. But I also think about things like buying random USB cables or devices off of Amazon and just fearing plugging them into your computer knowing that, "This just came from some random factory in China. Who knows if it's doing anything to your computer?"

[]00:49:39 JM: Right. Yeah. I mean, I bought like an Apple charger recently for my computer from a third-party seller on Amazon. Not only did it not work. Not only did like heat up mysteriously, but I was like, "Why am I doing this? This thing could easily have some kind of microphone in it." I'm connecting it to a power source. This could easily have a microphone in it.

[00:50:03] DG: Or a SIM card, or anything. It could be monitoring your network traffic, which is a very paranoid thing, but not unprecedented.

[00:50:19] JM: Yeah. Although network traffic doesn't go over a Magsafe cord, does it?

[00:50:24] DG: NO, but if you're plugging in, say, a USB cable.

[00:50:28] JM: Sure. Sure. Yeah. Okay. Yeah, sure. I mean, like in the case of if you have a USBC – Okay, I guess I have an old school laptop.

[00:50:36] DG: That's fine.

[00:50:37] JM: Okay. Well, we will get back to CamelCamelCamel, but I just want to ask you, any perspective on cryptocurrency blockchain related stuff, or just total nonbeliever in that category? What's your take?

[00:50:49] DG: Well, it's easy to look back with hindsight and think it was just a massive wealth transfer to Asia, Russia and China in particular. The real conspiracy theorist in me wonders if the graphics card manufacturers were involved somehow too, because they certainly saw the benefits of that.

[00:51:15] JM: All right. So you're believer in something related to cryptocurrencies. So as a privacy protecting technology, you are a dubious.

[00:51:23] DG: I think it's a little early to say. I think there is the potential for some – Blockchain itself could be useful in some ways. But for the most part, it just seems to have been used to take money from people so far.

[00:51:40] JM: Okay. Indeed. Okay, so advertising. You have advertising on your site now. How did you vet? Given your strong priorities around privacy, how did you vet the providers of advertising technology?

[00:51:58] DG: Well, we begrudgingly started using advertising. It's not our preference, but it came to a point where affiliate sales just needed a little – Not only did we need to kind of diversify our income a little bit, but affiliate sales were down and we needed to keep the site up. So the way that we vet companies is we only work with a select few, only the very biggest names, and we try to keep them headquartered in either the United States or Europe. Now, you're always going to have problems with privacy just because advertising is all about trying to figure out what you want.

Thankfully, ad technology has kind of improved now to where, say, if you have five ad partners, you don't have to load every single ad partner's code and give them the opportunity to look at the user before they decide if they want to buy that impression. Now it's –

[00:53:23] JM: Is header bidding?

[00:53:24] DG: Yeah. So before in the waterfall model, you would just load every single ad network one at a time until somebody bought the impression. Now, you have a much leaner model where you just send out a request for bid to everybody at once and they can't really load all these additional JavaScript for tracking and set a bunch of cookies even if they're not going to bid. So we're doing what we can to mitigate the privacy issues. We don't do any kind of ad blocker detection. People are welcome to block ads if they want, and we appreciate if they buy stuff through our site.

[00:54:11] JM: When you integrated with these ad networks that ultimately you don't know what they're going to run on your site. You don't know what their JavaScript blobs are going to do. Given your strong personal beliefs around privacy, did that cause you any cognitive dissonance?

[00:54:35] DG: Absolutely, and that's why we didn't do any advertising for quite a long time, but it just comes down to the thing we discussed before, where you have the choice to block ads or not. Really, we leave it up to the user.

[00:55:01] JM: Does blocking ads necessarily block all of the surveillant JavaScript blobs?

[00:55:08] DG: Yes. Well, I mean, I can't speak to the completeness of these specific ad blockers, but in my experience, the ad blocker will prevent access to certain domains or access to this well-known base JavaScript that could potentially load other scripts. So, say, the ad blocker blocks our code that kicks off the header bidding process. Nothing else gets loaded down the line.

[00:55:49] JM: Did you notice any increase in traffic after you integrated with ad networks?

[00:55:54] DG: Increase in traffic? No, not really. For some reason, our bounce rate in Google Analytics went down to nothing. I'm not sure why. Something to do with how the ads are loaded or making Google Analytics screw up.

[00:56:19] JM: So interesting. Any idea what might have caused that? Was that like the very first ad network, or was there a specific ad network that you integrated within and Google Analytics stopped working?

[00:56:29] DG: Google Analytics works. It’s just that one specific metric where it thinks nobody ever bounces. I don't really know why. I would like to know why, because then we could have real data again.

[00:56:44] JM: What does a bounce rate actually imply?

[00:56:47] DG: So a bounce rate is the percentage of users that make one page request to your site and then leave. So, say, somebody searches for a product. Finds our site and then leaves immediately. That's a bounce.

[00:57:09] JM: Could you imagine any ad fraud schemes that would be associated with that kind of drop in bounce rate?

[00:57:17] DG: Certainly possible, although it seems like that would not be so prolonged. This is something that's been an ongoing problem for quite a while now. I would hope for the ad network’s sake that they were not receiving fraud on the scale of years, but that's also not unprecedented.

[00:57:40] JM: I mean, do you know how ad networks work?

[00:57:43] DG: Oh! Fraud, all the way down.

[00:57:47] JM: Yeah, basically. I mean, tell me how criminalized or these ad fraud, or I should say ad tech companies. To what extent do they do play both sides of the table? The tinfoil hat wearing add fraud conspirator in me who would go hang out with the Bitcoin conspirator in you. I mean, I suspect that it makes sense for these companies to basically drive traffic, drive fake traffic to these same ads that they are displaying through their network. Am I totally crazy?

[00:58:29] DG: Well, first, I love this line of thought. I think, for a short-term scheme, that makes a lot of sense. It doesn't make sense for a company that wants to be around for a long time, because the companies that are doing the advertising, say, Nike is buying ads through some ad network. They're going to see that they're getting a bunch of valueless traffic from this network and they're not going to buy anymore.

[00:59:01] JM: Assuming they audit it, and assuming it wasn't purchased through like a third-party marketing agency, which it often is.

[00:59:09] DG: That’s true. It could be that everyone is working together to defraud the big corporations.

[00:59:16] JM: Which is actually the most hilarious part of it.

[00:59:20] DG: Sort of Robin Hood situation.

[00:59:22] JM: It absolutely is. It absolutely is. It's like the corporations that provide us "free services” are taking their money from basically the dumber corporations. My words, not yours.

[00:59:37] DG: One hopes that they’re auditing things, but –

[00:59:40] JM: Oh, yeah. I’ve spoken with the auditors.

[00:59:46] DG: From a beach in – Some tropical beach somewhere, I'm sure.

[00:59:52] JM: Well, no. I mean, you might actually like these episodes. I’ve done a couple shows with these guys, Method Media Intelligence, and they – I don't know. They're profiteers like the rest of us. But, I mean, they basically built their business off of being like, “We are the ad fraud intelligence guys. We tell it like it is,” and just like the levels of fraud that they have kind of introduced to me, I'm just like, “Jesus! This is a total mess.” What’s so funny about it is how much anti big tech rhetoric there is these days, and nobody talks about advertising fraud.

[01:00:29] DG: No. Unlike privacy, it’s not something that really impacts the end-user. So why do I need to care about it? It's big corporation getting what they deserve.

[01:00:45] JM: Right. Okay. That's funny. So we’ve been talking for an hour. I want to begin to wrap up. Very interesting company, very interesting, basically, trajectory through your tech career. Did you ever have a moment where you were like, “Maybe I want to do like a startup. Maybe I want to raise money. Maybe I want to build a company, or maybe I want to like get CamelCamelCamel acquired,” or have you always just thought of CamelCamelCamel as just this like fun sandbox. It's like your personal side business you get to run with your friends. Has it always been scoped to kind of the “small business”? I’m sure it’s a large business, but a “small business”?

[01:01:28] DG: Well, I never really had a desire to blow it out to have tons of employees or anything, and early on it was certainly just something that was fun to work on. We didn't plan for it to be a business.

But around the same time we were coming up was when, say, Google ventures and Y Combinator were coming up as well, and having friends who were in technology during the original .com. Of course, me as an early twenty something was always just enamored by the idea of, “Wow! We could get acquired by Google or whoever, and life would be amazing,” and we did pitch actually three camels to Google ventures at one point, and that went about as well as anyone could have predicted. Just not a big enough idea for them.

But, over time, the naïve shine of the startup culture kind of faded away and we’ve been really happy that we didn't take funding, because we get to determine our own future and we’re not beholden to anyone, whether that's limited some opportunities potentially, but we’re really happy with the choices we’ve made.

[01:03:08] JM: Daniel Green, thanks for coming on the show. It’s been actually really, really fun talking to you.

[01:03:13] DG: Thank you for having me, and likewise, I’d love to do just the whole conspiracy theory hour next time.

[01:03:20] JM: Oh, man! I mean, that's actually what Software Engineering Daily is all about. It's all under the guise of being about boring databases, and infrastructure, and stuff, but it's a Trojan horse for the conspiracy theories.

[01:03:35] DG: I love it.

[01:03:36] JM: All right, man. Talk to you later.

[END OF INTERVIEW]

[01:03:42] JM: This podcast is brought to you by wix.com. Build your website quickly with Wix. Wix code unites design features with advanced code capabilities, so you can build data-driven websites and professional web apps very quickly. You can store and manage unlimited data, you can create hundreds of dynamic pages, you can add repeating layouts, make custom forms, call external APIs and take full control of your sites functionality using Wix Code APIs and your own JavaScript. You don't need HTML or CSS.

With Wix codes, built-in database and IDE, you've got one click deployment that instantly updates all the content on your site and everything is SEO friendly. What about security and hosting and maintenance? Wix has you covered, so you can spend more time focusing on yourself and your clients.

If you're not a developer, it's not a problem. There's plenty that you can do without writing a lot of code, although of course if you are a developer, then you can do much more. You can explore all the resources on the Wix Code's site to learn more about web development wherever you are in your developer career. You can discover video tutorials, articles, code snippets, API references and a lively forum where you can get advanced tips from Wix Code experts.

Check it out for yourself at wicks.com/sed. That's wix.com/sed. You can get 10% off your premium plan while developing a website quickly for the web. To get that 10% off the premium plan and support Software Engineering Daily, go to wix.com/sed and see what you can do with Wix Code today.

[END]