EPISODE 581

[INTRODUCTION]

[0:00:00.3] JM: Starting an internet business is harder than it should be. You need to incorporate, create an operating agreement, set up a system to accept payments and many other seemingly straightforward tasks. In the 1990s, this was how it felt to set up anything on the internet. You always had to stand up a web server on your own infrastructure before you could get to the interesting part, which was building an actual product. But the popularization of cloud computing made it easier to standup a server. Because of that lower activation, millions of applications and thousands of software businesses got started, but the activation energy required to start the business remains higher than necessary. It feels like standing up a web server in the 90s. There's lots of tedium and reinventing the wheel that has been done by people before you. This is the motivation behind Stripe Atlas, which is a project to simplify the process of starting an internet business.

Patrick McKenzie works on Atlas at Stripe. He was previously on the show to discuss his experience building a small software company or a series of small software companies after leaving a large corporation, and his name has become synonymous with the modern phenomenon of this small software entrepreneur. He has been riding about this topic for over a decade at kalzumeus.com. It was great to talk to Patrick once again about internet businesses, and I'm excited to see Stripe Atlas become something huge.

We've done four other shows about Stripe as well as other financial technology, like blockchains and automated trading. If you're looking for all 700 episodes of Software Engineering Daily, you can check out our apps on the iOS or android app stores. You can listen to all of our episodes at softwaredaily.com and you can find all of our episodes without ads if you become a paid subscriber. If you're interested in subscribing, you can do that at softwaredaily.com. Thanks for listening, and let's get on with the episode.

[SPONSOR MESSAGE]

[0:02:08.3] JM: Azure Container Service simplifies the deployment, management and operations of Kubernetes. Eliminate the complicated planning and deployment of fully orchestrated containerized applications with Kubernetes. You can quickly provision clusters to be up and running in no time while simplifying your monitoring and cluster management through auto upgrades and a built-in operations console. Avoid being locked into any one vendor or resource. You can continue to work with the tools that you already know, such as Helm and move applications to any Kubernetes deployment.

Integrate with your choice of container registry, including Azure container registry. Also, quickly and efficiently scale to maximize your resource utilization without having to take your applications off-line. Isolate your application from infrastructure failures and transparently scale the underlying infrastructure to meet growing demands, all while increasing the security, reliability and availability of critical business workloads with Azure.

To learn more about Azure Container Service and other Azure services as well as receive a free ebook by Brendan Burns go to aka.ms/sedaily. Brendan Burns is the creator of Kubernetes and his ebook is about some of the distributed systems design lessons that he has learned building Kubernetes. That ebook is available at aka.ms/sedaily.

[INTERVIEW]

[0:03:42.5] JM: Patrick McKenzie, welcome back to Software Engineering Daily. You work on Stripe Atlas. Thanks for coming back on the show.

[0:03:49.2] PM: Yeah, thanks so much for having me.

[0:03:50.2] JM: Last time we spoke we were talking about starting small internet businesses of which you have began several in the past. You were just starting to work on Stripe Atlas at the last conversation we had. Give the listeners an overview of your business background. How did you wind up at Stripe?

[0:04:09.7] PM: Sure. I usually start where I had graduated college and moved to Japan with the idea that I was going to get better at business Japanese and then get a job in the US

software industry in a Japan-facing rule at a place like Microsoft. I had it written out and stamped on my wall. I wanted to be the project manager of MS Excel Japanese version. That was my terminal career goal. Then life is what happens to you while you're busy making plans.

Worked at a Japanese organization for a few years and got little tired of working on a traditionally managed Japanese company and during my nights and weekends, first I ran a World of War Craft guild and then I decided to do something with less dragons and better loot and start a software business. So I build a company called Bingo Card Creator which made bingo cards for elementary school teachers and it was originally just like strictly a hobby to take at the time that had been used by well, but eventually grew into something that was much more than a hobby. I don't remember the exact day. It would have been in the late 2009 where I was crunching very hard at the day job and had worked 19-hour a day. Checked in to a hotel to sleep for 4 or 5 hours before I had to go to work the next day, and I woke up in the morning and checked my email. I had a kindle as I was doing back in those days, and I had gotten enough PayPal that you've got money notifications overnight to tell me that I had made more money while sleeping than when working the 19-hour a day previous to that. It was like, "Deary me, What am I doing?" So that was probably the exact minute of which I knew that I was quitting the day job. So that was good.

I quit the day job. I spent the next couple of years doing a combination of — I built a software consultancy that was mostly helping out other SaaS companies with marketing and sales, which seems to be the thing that I have a comparative advantage on versus most. So I had people work in software. At the same time, I was building up an appointment reminder which was a SaaS business which [inaudible 0:05:55.0] appointment — Well, appointment reminders. I'm terrible [inaudible 0:05:58.5]. Automated SMS phone and email reminders to the clients of professional services, businesses.

For example, if you have a doctor appointment at 2:00, you would get a call at 8:30 am that morning saying, "Are you coming to the doctors? If so, press one. If not, press 5." If you needed to cancel the appointment on short notice, we could have a phone automatically call up the doctor's receptionist and get them to rebook that slot so that they don't lose revenues that they can never get back when they lose a service provider time.

So I ran that business. I sold Bingo Card Creator and started in 2006, sold it in 2015, sold Appointment Reminder in 2016, consulted actively for a number of the years in there. I did a company called Star Fighter in 2015 and 16, which didn't end up working out. This is with two of my friends, Cisco founders, Thomas Ptacek and Erin Ptacek.

I had this idea that we could make engineering hiring tranformatively better by building game that people could play for no reason at all, and the game give you a hard engineering problems that you would solve by programming.

We would take a look at the folks who did well playing the game and say, "Clearly, you're very good at this. If you're happy doing what you're doing, that's awesome. But if you're not happy doing what you're doing professionally, perhaps we could introduce you to a firm that could make better use of your services." If one of our clients took an introduction to one of our players and successfully hired them, we would get a contingency recruiter fee out of that. Sadly, that provided reasons that didn't end up working out.

[0:07:27.5] JM: That was a bellweather of a lot of different hiring companies that end up working out too, or at least going the right direction with that company.

[0:07:36.1] PM: Yeah. I think the core hypothesis we had for it was substantially accurate. Hiring in engineering is still broken. It's still a multi-gazillion dollar business if someone does it correctly. I think our fundamental approach was pretty sound to it, but a combination of sending a little too much time on the development versus getting it out into the market and some other shoes that are probably a little outside the scope of this podcast.

[0:07:58.7] JM: Yeah, right.

[0:07:59.1] PM: Limited to that particular version networking.

[0:08:01.5] JM: I think people at this point can understand why you ended up working on Atlas, because you were tinkering on these niche business ideas before the small software business idea became more popularized before the Indie Hackers movement and before people started realizing how much money there could be in these niche SaaS businesses. There's a variety of

internet business types and most of the coverage goes to these venture back startups, but there's also small businesses like Amazon sellers. People have built entire companies around selling things on Amazon and then maybe they have small software products that are linked in to those. Then there are the Indie Hacker type of businesses which it's been awesome to see Courtland shed more light on.

But there's really a large gradation of these software businesses and everyone knows that figuring out a business model and building a product is difficult, but there really is a wide range of markets and market sizes to go after. But I think what we're going to get to today is a discussion of why it's still so hard to set up a business.

So what people may not know is that setting up a business, like legally, doing the legal incorporation, issuing shares, that process is actually difficult. I would say it's as difficult as finding a business model and building out a product, but it is not the kind of thing that should be difficult at all, because in that case there's nothing technologically complex going on. There shouldn't be anything strategically complex going on. This stuff should just be worked out. It should be taken care of for you. Yet it's still as hard as spinning up a server in the 90s when you had to figure out load balancers, and like it was hard to configure your database and all these other things that were way harder than they needed to be. Why are we still at this point today where spinning up a small business or a large business, of course, is way harder than it needs to be?

[0:10:13.0] PM: I think you can think about the technology of operating a business in similar fashion to the technology of operating technology to coin a phrase. We're in the tail end of the OSS moment right now where in like the first generation of internet businesses, it's like you want to sell something online? Great. Step number one; write a web server in C++. That was a really rough, rough place to be in, not because writing a web server is the most complicated program that you could possibly write, but because it is a relatively complicated program and it gates everything else that you want to do.

So part of the Cambrian explosion that we've seen in a lens offer businesses and online business of all types has been enabled by OSS and by other platforms like, say, Amazon Web Services. The availability of other OSS that's available at — OSS services that are available

with a relatively low barrier to entry including OSS and that allows people to focus less of their energy and just getting the infrastructure working and more of their energy on actually producing in the business value and getting that in the hands of customers.

So to your point, writing the operating agreement for an LCC shouldn't be an earth shattering novel work of creation, but historically the way the regulatory system works and the legal system works is that they assume that essentially all work like that is bespoke and done every time by relatively highly priced professionals and it's not something that is just perfected and tweaked and then are reused off-the-shelf.

What we're trying to do on Atlas is to take the infrastructure of building businesses and make it more like OSS as - such that a company that is starting up can say, "Look. I want a plane [inaudible 0:11:55.9] corp, the sort of thing that an investor can look at and say, "Yup, I would invest in that," and just to have that ready to go at the click of a button, just like you can have an Ubuntu image ready to go at a click of a button on AWS. So you could spend less of your scarce entrepreneurial bandwidth, the stuff that doesn't add value, like your articles of incorporation for a company. Something that I know lawyers. I love some lawyers, and they will tell you that your articles of incorporation are very important. That's probably true, but they're probably not a source of a huge amount of marginal, importance contingent on you getting one that is functional. So have an easy way to get one that is functional and then spend your [inaudible 0:12:32.4] bandwidth budget and your risk budget and your unique decisions budget in places that uniquely create value for your customers.

[0:12:39.7] JM: Right. Because once you find product market fit, then it's fine. Spend time and money on lawyers. Get your operating agreement revamped. Get your articles of incorporation revamped as necessary. But until then, those things are not worth spending time that people often end up spending on them. They end up engaging with lawyers and accountants and reinventing the wheel in places where it's not necessary.

Now, of course, there are going to be naysayers to this argument, because by lowering the barrier to entry for — Let's take cryptocurrencies, for example. So we're seeing some serious downsides to the barrier to entry being lowered for issuing security. We had the ICO craze that made it easy for people to create scams that look like serious operations. People took it

seriously, this idea that, “Oh! This is a security that has been issued and I can buy it on this sketchy exchange.” So I’m not drawing a direct comparison to issuing a coin and getting your articles of incorporation drafted, but I’m trying to raise the devil’s advocacy of are there good reasons why it is not extremely easy to set up a C-corporation?

[0:14:02.4] PM: Well, there’s a spectrum of things here. Let’s say there’s a very little good reasoning, that it is hard to set up a C-corporation as a legitimate entrepreneur or you have a business like all startups, there might be certain amount of risk involved, but that you’re fundamentally going to make a go of it. It’s just purely a task on your time and attention.

There’s the hoop jumping barriers that are between you and having a company set up are not really for the benefit of protecting the public from you. Delaware doesn’t do a review process and say, “Okay. You’ve convinced us that you are on the level and we will award you with a C-corporation.” They will give one to literally anybody who fills out the correct paperwork. It’s just that figuring out what the correct paperwork is is harder than it needs to be.

So democratizing the technology of producing the correct paperwork in a time efficient fashion and saving lawyers cycles for places where the lawyer can uniquely help you to avoid legitimate risks to the business just enables people to produce value producing companies better.

[0:15:06.1] JM: So what kinds of paperwork does a new corporation need to fill out to get set up? What are you trying to automate with Stripe Atlas? Take me through the different things that you needed to automate to get the MVP of Stripe Atlas up and running.

[0:15:21.4] PM: Sure. I guess the first thing to point out is that there is a garden of branching paths in attempting to create in a sort of company and there is a lot of decisions that you have to make. As an entrepreneur who might know how to get a web application working, but probably doesn’t have an MBA. A lot of the decisions are highly non-trivial. For example, do you want a C-corporation versus an LLC? Given that you want a C-corporation, do you want to incorporate in Delaware or your home state or some third state?

Part of what we did is we just were able to make some opinionated choices, and if you’re building a C-corporation, you should just build in Delaware rather than looking at the other 50

jurisdictions that could incorporated C-corporation within the United States, and there are some reasons to go into that, which can and can't go into a huge amount of detail.

The long story short is that a Delaware C-corporation is adequate to almost all purposes that you would put a C-corporation, and given that corporations in various states are more or less fungible for each other for most purposes, but then investors are much more comfortable in investing in Delaware C-corporations for boring legal reasons. It's easiest to just go with the one that preserves a lot of optionality for you in the future.

Also, it's a way to signal to entrepreneurs. You will have so many consequential decisions over the next five years of your life, and the state that you incorporate in is not one of those consequential decisions that is going to determine whether this business is going to be successful or not. Therefore, don't spend all that much time thinking about. Just go with the sensible defaults and that will take you as far as you need to go on this part of the decision and spend your energy and your brain sweat on harder questions, like who should my cofounder be, or what market should we go after? What marketing approaches are we going to use in the next 6 weeks to close some deals, etc.

[SPONSOR MESSAGE]

[0:17:15.0] JM: Users have come to expect real-time. They crave alerts that their payment is received. They crave little cars zooming around on the map. They crave locking their doors at home when they're not at home. There's no need to reinvent the wheel when it comes to making your app real-time. PubNub makes it simple, enabling you to build immersive and interactive experiences on the web, on mobile phones, embedded in the hardware and any other device connected to the internet.

With powerful APIs and a robust global infrastructure, you can stream geo-location data, you can send chat messages, you can turn on sprinklers, or you can rock your baby's crib when they start crying. PubNnub literally powers IoT cribs.

70 SDKs for web, mobile, IoT, and more means that you can start streaming data in real-time without a ton of compatibility headaches, and no need to build your own SDKs from scratch.

Lastly, PubNub includes a ton of other real-time features beyond real-time messaging, like presence for online or offline detection, and Access manager to thwart trolls and hackers.

Go to pubnub.com/sedaily to get started. They offer a generous sandbox tier that's free forever until your app takes off, that is. Pubnub.com/sedaily. That's pubnub.com/sedaily. Thank you, PubNub for being a sponsor of Software Engineering Daily.

[INTERVIEW CONTINUED]

[0:18:57.4] JM: Yeah. So my personal experience with this is you could set up a C-corporation, you could set up an LCC, you could set up a S-corporation, you could set up a B-corporation, but for the most part, early on, you're not going to even care which one of these you're setting up, because probably your business has not even gotten off the ground yet. You're probably not wildly profitable and many of the tradeoffs between these designations have to do with how cash flows propagate from the organization to the holders of equity within the organization and none of that really matters if you don't have cash flows from the beginning. So you might as well just check the box that gives you the optionality to configure how those cash flows get directed later on in the future and at least sets up the organization for the lowest barrier to entry to fundraising, because most likely your business is not — I mean, if we're just talking about the raw population distribution of how companies work out, you're probably not going to have a wildly profitable business and you're either going to foil or you're going to need to raise money, which makes sense to configure it as a C-corporation.

[0:20:14.2] PM: I think that one of the infelicities about how corporations have historically worked is that you sort of need to be able to predict the future. If you know that you're going to take investment as of day one, then clearly you should be C-corporation, and then if you think that you will never take investment and you will never have more than one owner, then there is probably a fairly good case for you being an LLC. But you need to know that on day one, and that's relatively difficult.

We've got some stuff coming down the pipe which will hopefully allow people to maintain more of the optionality with regards to what their corporate form is, but listen to the space in the near future or rather in the space, stripe.com/atlas. That being said, there are some differences I the

various entity types. One of the things that we do is we try to educate people on the difference between the various entity types so that you can make a good enough for you right now decision without having to spend a lot of time with lawyers and accountants playing their specialized knowledge to a question that doesn't really require all that much specialized advice to get right.

[0:21:14.1] JM: There are a lot of products at Stripe where engineering is complex. The stripe antifraud system, for example. That seems like an engineering challenge. Atlas, to me, looks like more of a challenge around customer service, product management, integrations, regulatory compliance. It's not as much about really hard engineering problems as it is about diplomacy and building a relationship with your integration partners, like accountants and lawyer-based services. Is that accurate? Because you're kind of in charge of the Atlas project. Is that your experience?

[0:21:56.0] PM: I'm not formally in charge of the Atlas project. I work on the team, like many smart people work on the team. The manager of it is a gentleman named Taylor, and then we have Alex who does the engineering management, but be that as it may. I think there are some hard product-y kind of decisions on how do you take a user who might not have a huge amount of background understanding of like what the issues are at play and get them through in a corporation of a full, real-honest to God company in a fashion where they understand everything that's going on at their behalf and that they feel comfortable doing it.

Some of that is from design. Some of that is from good product decisions. You're right. It does not processing on terabytes of data every night so that you can do X-thousands of transactions per second through a machine learning network. That's not the nature of the challenges we have on Atlas.

[0:22:44.4] JM: Today.

[0:22:44.9] PM: Today. You can imagine all sorts of weird things that could happen as time goes to infinity, but there are substantial amounts of engineering not as it is normally understood with regards to how do we route users to experts in the fields that they need right now. How do we handle a customer to support for users? The expectation among the users for Atlas is that they

can come to us with any question under the sun about their businesses. We'll do our level best to answer that if we're possibly capable of answering it. We're not a legal firm. We can't provide legal advice for non-accounting firm. We can't provide accounting advice. If you say, "Hey! I'm working on an investor pitch. Can you help me with that?" Our default answer to that is, "Yes, we would be happy to help you on your investor pitch." That's a different flavor of customer service than we've traditionally offered at Stripe where usually it's like, "Okay, we're super comfortable helping you through like your API integration, but it's not like we have an existing process that says, "All right, take flawed investor pitch. Apply transformation for it and get out better investor pitch." So we've been sort of building up those muscles over the last year, two years, and it has been a very interesting thing to work at.

[0:23:53.5] JM: Well, this is what I love about Atlas, is it's like you put paper mache around something that is deeply complicated and it's not like you're fixing it with engineering quite yet. I mean, you're fixing it with internal organizational processes. Like Stripe is a well-run organization so that you can — Stripe is built to be able to handle edge cases and complex customer service issues because of the way the culture has developed. Atlas is the perfect type of product to develop because you can just paper over a highly complex and unstructured problem by just putting smart people in the middle and then overtime the smart people can talk about, "Okay. Here's the thing we're finding comes up over and over and over again. Let's build an automated product around it."

[0:24:47.3] PM: I think the thing that you'll hear over and over again in product management, whether you had a company like Stripe or whether you're just starting out is do things that don't scale to start out so that you can learn about the problem domain. So back in the day, there was relatively little engineering in Atlas at all. It was a form that would add your information into a database and then there would be actual people at Stripe like pushing buttons on the fax machine to get you incorporated, and we're substantially more automated these days. Sort of like seeing what the process looked like on a sort of an artisanal scale and then pave in the compass was a core way that got us to where we are right now. We have a team of a dozen or so smart people working on Atlas every day and we run into a certain amount of issues which are present over a large portion of user base and which are relatively credited for us. Sometimes we handle those by just rolling up our sleeves and applying less elbow grease, and then sometimes after applying the elbow grease for a few times, it's like, "Okay. Where can we

make intelligent investments on the engineering side to solve this on a more scalable fashion such that we can spend our elbow grease new issues?"

[0:25:56.1] JM: There's all these required legal and accounting and business knowledge for building Atlas. By the way, I think we've put the emphasis on the legal side of things, but Atlas also in the process of signing up, you get a bank account and you get integrated with Stripe. I have a business that I use with Atlas and when tax season was coming up, Atlas was notifying me, "Hey! You need to file your taxes and here are some ways where we can potentially help you." Accounting is a core competence.

[0:26:33.0] PM: Yeah, we're happy to introduce folks to accounting firms that are appropriate for the stage that their business is at and to have them use those accounting firms to file their taxes. One of the perennial issues for us is that we are not ourselves lawyers and we have not ourselves accountants, so we can't give accounting or legal advice directly, but we can give information to people and give them the pointers in the right direction both in terms of like specific firms that they could engage on their behalf or additional things to research such that they can handle their situations for themselves.

[0:27:02.4] JM: There are some froth in this space. Founders have access to things like UpCounsel, which is Uber for lawyers. There's also Uber for accountant type services where you can find an accountant on demand. I have had mixed experiences with things. Sometimes I get really good value of them. Other time I waste so much time and money and I just say to myself, "I wish I would have gone to an actual lawyer or an actual accountant." It seems less clear to me that these services are the right solutions in contrast to something like Task Rabbit or Uber where there is more fungibility to the market. Is that your experience as well with these like Uber for lawyer, Uber for accountants?

[0:27:54.2] PM: Well, we do partner with UpCounsel and —

[0:27:57.0] JM: By the way, I love UpCounsel. I've had very good experiences with UpCounsel. I maybe had one marginal experience. But yeah, just to caveat, I do love UpCounsel in general.

[0:28:07.5] PM: Oh, awesome. Glad to hear that they've worked out well for you. Broadly speaking, I think that any sort of service which encapsulates like an underlying professional that's providing the service has an issue with scoping. For example, the scope a service provided by Uber is fairly easy to understand. You are at point A. You want to go to point B. You don't want to die in the middle. Any person that can execute on that well will largely execute to first approximation as well as anyone else executing on it. That's not true of the vast field that could be law. It's like there's a fractal nature to it where there is infinite variability, and then if you zoom in on any like particular sub-field, there's infinite variability within that sub-field.

Part of the nature of Atlas is that if you can't make law less complex than law is can you constrain the nature of the problem space such that any provider that you tap is always looking at a more constrained problem. One of the ways that we can strand this is by saying, "Okay. Atlast C-corporations will invariably be Delaware C-corporations and they will invariably be using this paperwork that we've provided for them." So you don't have that point with a lawyer of, "All right. Try to go from the entire universe of world states that a company could possibly be in," to select down to the one they're actually in and then start the practices of law. Say, "Okay. We can prepare you with regards to all of the companies you do business with from Atlas. They will be in this particular state." Start from that foreknowledge of where the problem is going to be and hopefully that makes things more efficient for the service provider and better for the user and that they don't have to redundantly explain their situation as a lawyer and/or accountant and every other service that they integrate with.

[0:29:52.6] JM: Tell me more about how product development works within Atlas. I know maybe you can't comment specifically on what is being built right now, but it sounds like this process of having fax machines and doing everything manually and then figuring out what the automate is, a core competency there. Is there anything else interesting you can comment on about how the product development process works broadly?

[0:30:20.8] PM: A part of it looks a lot like [inaudible 0:30:22.7] at any company. You spend a lot of time talking to our users both with regards to inbound requests from them to our user operations team and also proactively just reaching out and saying, "Hey! What's working in your business right now? What are your core challenges? What you feel like you could be helped on

that you're not getting helped on?" Then we attempt to do experiments that prove out that we could be useful with regards to particular things.

An example of this [inaudible 0:30:47.6] is just having smart people around that are dedicated to helping our customers out. We knew that a lot of customers were looking at trying to get investment for their companies, and we said, "Okay. Well, YC season is coming up. I personally know a little bit about like what YC is looking for in applications. Let's do something crazy. We're going to email everybody on Atlas and say, "Hey, if you want to get YC investment, just hit reply on this email. Send us a link to your application and we'll take a look at it for you and maybe help you write it such that your narrative lows a little better."

So I got an arbitrarily large number of people hitting reply on that email and there was no technology built out. There was no process. There was no triage workflow in ZenDesk. It was went straight to my inbox and then I worked a little too hard for little too long. Getting replies out to everybody, and that worked out pretty decently the first time. Then six months later we're like, "Okay. YC thing worked out pretty well for users, but it didn't work out exactly pretty well for me. Can we get a little bit more process around this to random example?" We just told people, "Yeah, just get us your application," and if you give an unbounded instruction like that to a large group of people who have technical skills that's all over the map, you'll get surprising for you file formats including Google docs. I copied-pasted all my stuff into markdown and then attached it to the email. My personal favorite —

[0:32:14.2] JM: Exe? You get an exe?

[0:32:16.1] PM: I got iPhone screen grabs of someone's application that was open on their Macbook. It was my personal favorite for attempting to annotate an application. So the second time around, we wised up and we're like, "Copy your answers into a Google doc, and then share us the Google doc." We can have folks on our user operations team verify that the Google doc is correct prior to sending it somebody like me to get feedback on. That we do that sort of thing. That allows us to provide the same level of service, but maximize the amount of my time that is spent talking to people about deep issues in their business or about positioning it for the benefit of the YC reviewers and a minimal amount of time attempting to figure out like, "Hey, you sent

me this file, but it didn't actually open. Can I ask that you resend it in MS Word 2017?" yadi-yada.

[SPONSOR MESSAGE]

[0:33:09.6] JM: Software workflows are different at every company. Product development design and engineering teams each see things differently. These different teams need to collaborate with each other, but they also need to be able to be creative and productive on their own terms. Airtable allows software teams to design their own unique workflows. Airtable enables the creativity and engineering at companies like Tesla, Slack, Airbnb and Medium.

Airtable is hiring creative engineers who believe in the importance of open-ended platforms that empower human creativity. The mission of Airtable is to give everyone the power to create their own software workflows from magazine editors building out their own content planning systems, to product managers building feature roadmaps, to managers managing livestock and inventory.

Teams at companies like *Condé Nast, Airbnb and WeWork can build their own custom database applications with the ease of using a spreadsheet. If you haven't used Airtable before, try it out. If you have used it, you will understand why it is so popular. I'm sure you have a workflow that would be easier to manage if it were on Airtable.*

It's easy to get started with Airtable, but as you get more experience with it, you will see how flexible and powerful it is. Check out jobs at Airtable by going to airtable.com/sedaily. Airtable is a uniquely challenging product to build and they're looking for creative frontend and backend engineers to design systems on first principles, like a real time Sinclair, collaborative undo model, formulas engine, visual revision history and more.

On the outside, you'll build user interfaces that are elegant and highly customizable that encourage exploration and that earn the trust of users through intuitive, thoughtful interactions. Learn more about Airtable opportunities at airtable.com/sedaily. Thanks to Airtable for being a new sponsor of Software Engineering Daily and for building an innovative new product that enables all kinds of industry to be more creative.

[INTERVIEW CONTINUED]

[0:35:29.5] JM: The scope of what Stripe Atlas could become eventually is quite wide, and I wonder what led you to choose something like that that is kind of bandwidth intensive, like asking people for their YC apps so you can help review them and improve them. That's time consuming, and for all the choices of product development bandwidth, you could allocate towards. You went down that path. Is there an overall KPI that you're going towards that in trying to fulfill that KPI you are like, "Oh! If we solicit people for YC applications, that will increase that internal KPI, or was it just like, "I'll just do this on a flyer and see what happens."

[0:36:13.3] PM: When I was running my own software companies, the basic business model was that I was running a vending machine over the internet where I did not want to talk to you. If ended up talking to you for a customer service request or something, I treated that as bug. Older and wiser these days, I think that is not the way to do things.

So sometimes when you're doing things that don't scale, you should be doing things that are bandwidth intensive specifically because they're bandwidth intensive. Spending a lot of bandwidth on talking to customers about what they're current state of their company is and what their aspirations are for the future on it and how they expect to get to that future is an extraordinarily useful thing to do for customer development for us.

Obviously, we want to provide value from the perspective of materially improving their YC applications, getting them the investment they want, etc., etc., but it's also an excellent way for us to learn in a very high bandwidth fashion about where our users are right now and what else we should be building for them. When we have 100+ conversations with people, we can start up picking up on like reoccurring themes of what are the specific problems that someone is having. What are the industries that people are very interested in? What are the business models they're building up and make sure that we are building our product in a way that's worth those use cases that people seem to want to apply the product to.

[0:37:28.3] JM: What has surprised you in this high bandwidth scope of exposure to newer companies?

[0:37:35.7] PM: So many I think surprised me. There are some unsurprising surprises. So the — Unsurprising surprise is the one where if you were to ask someone before they do something, predict the surprises that you're going to say. You would be able to predict the surprise, but that you would not be able to predict the magnitude of the surprise. If you're to ask me, you're going to talk to a hundred random software companies. Predict the surprise [inaudible 0:37:56.4]. I would say I bet they're going to be surprising for idea of what people are doing to make money on the internet. That's true.

But then you actually talk to a hundred software companies and no matter what your model is for how many ways there are, just going to get how many ways to create value over the internet. You are under predicting the diversity of things that people are doing and under predicting the sheer level of creativity that people can bring to what they're creating in the world. Honestly, that's one of my favorite parts of doing this, is that we enable folks to do things that are not just new and exciting, but that you couldn't even predict of being possible without actually watching someone just go out and do it, and, "Oh my goodness! Are people capable of going on and doing it?" Obviously, for any given cohort startups, some work out pretty well. Some will work out less well, but the companies that are most successful have a surprising level of success.

Often, if you have a model for given what I know about a company on day one, who is going to be the most successful two years. We've had the opportunity to walk along with a selection of thousands of companies over the first year of their life, two years of their life. The ones that end up really, really compelling at the end of two years are often not the ones that look like, "Obviously, this will be a compelling business," as of the day one. That's been an —

[0:39:14.3] JM: What do you think is the X-factor there? What's the thing that you're not able to break? Is it the size of the market that you just didn't realize that, "Wow! Actually, there was a huge market for this market," or is it the quality of the people that you're unable to audit from just a YC application?

[0:39:31.5] PM: I think there's a variety of factors. I continue to think that people are surprising. It might be that short story version of my life, short version story of my life English. I don't speak it anymore. Sorry guys. No matter how much trust you have in the ability of people to produce outstanding work, you will occasionally meet folks who exceed every rational expectation and

many irrational expectations about the sheer amount of things that they accomplish within a short amount of time and short of seeing someone being amazing. I don't have 100% accurate algorithm for predicting who will be amazing.

Sometimes the founder sees something about a business model that you don't. Sometimes it's due to one being underinvested in this space, like you're not the exert, they are, and therefore they understand it better than you. Sometimes you have someone come to a space with the beginning mind and do something that's totally off the wall from the perspective of what the "best practices" would be there and succeed wildly. That's - both very interesting things to see.

[0:40:32.7] JM: That massive productivity where people surprise you with how productive and how much value they're able to create in a short amount of time, is that due to the increasing leverage that people get out of software tools or do you think it's just people pushing themselves hard? I mean, I sometimes think just the leverage that you can get out of even just like the product suite in AWS is massive. AWS, I think, is the most underrated advancement relative to how much value it has provided to the world. I honestly think that's probably the most underrated advancement in technology.

[0:41:12.7] PM: AWS is certainly a hugely, hugely important source of leverage for entrepreneurs that are using it, and particularly heavily technical entrepreneurs that are doing things that are ambitious. Even for folks who are running of the single server, I think it's probably the best solution for getting a single server up and running. I do think that 2018 is the best year in the history ever to be running a business, particularly technology business. There are massive sources of leverage. The internet as a distribution system, the individual channels on the internet, everything from Google search, to the app store, to the YouTube is a probably underrated distribution channel to the networks that are developing between consumers of various services.

All these things are a wind in the sails of anybody who's getting started these days. I think you can get substantially more accomplished as a small team due to the confluence of great hosting platforms like AWS, other sorts of infrastructural platforms, like Twilio or like, knock on wood, Stripe, being able to track people money kind of useful.

Things like OSS technologies, so you're spending the majority of your time building things and actually building things that customers will use rather than building out web servers or firewalled rules or all the other things that need to be done to have an operational business but don't really surface any sort of unique value to it.

Separate from the amounts of leverage that's available, I do think that there are productivity of people. There's probably some general factor of productivity, but it's probably not the same for a given person across all the possible situations they could be in. I think some people have come from backgrounds where they are a cog in a large machine and they just find out that they're not constitutionally well-suited to being a cog in a machine, and that when they are sort of unleashed and allowed to make their own decisions, that they turn out to be pretty good at making decisions and pretty good about motivating themselves.

I think that the larger edifice of industrial capital has produced a few technologies for getting productivity out of people, like bosses where like the institution of bosses can be one that is useful for the vast majority of people globally. There are at least some people who don't benefit from having a boss as much as other people do, and they might select — What's the opposite of the word benefit? Their productivity might diminish from having a boss. Whereas if you are able to give them the technology to say, "Hey, you can run your own company and be as productive as you are capable of being," that they will rise to the occasion.

[0:43:39.7] JM: Yeah, give them Google calendar. Just like have their calendar be their boss. I sometimes think about like my Google calendar is like the scheduler, like the operating system schedule for my life and it's just like allocating tasks to me to do throughout the day and with variable reliability of executing those tasks. But it does in some sense function like a boss. Talking more about small business, I know you were at Micro Conf somewhat recently and you had some tweets around that time, and I think Micro Conf is a real-world manifestation of the state of small internet businesses in some regard. Tell me about what you saw there and what you felt in the atmosphere around talking to people about their small internet businesses.

[0:44:32.0] PM: Micro Conf is sort of my home away from home. One of the things that I think is very important as a small entrepreneur is to find a group of people. When I was living in the middle of nowhere in central Japan, I would get on the bullet train for three hours to go out to

Tokyo just to meet a community likeminded entrepreneurs in Tokyo. I think Micro Conf is the single densest population of likeminded small software businesses anywhere in the world.

So we converge Las Vegas once every year or so and just to have a few days of talking shop. Everything from like micro mechanics of running a business, “Hey, I’m thinking of doing a drip email campaign for my info-products business. How should I set that up? What’re the kinds of emails I should send to help me with copywriting?” through like, “Okay. I’m thinking of moving upmarket in my SaaS business. Am I ready to move upmarket? What should my practice strategy start to look like? Does this impact how my sales model should work?” I think that entrepreneurs at Micro Conf are — Disproportionately, they are — They have a good integration of their business works with the rest of their priorities in life. So everything from how do I protect my family from the emotional ups and downs that are involved in running a business. If that’s important, do I need to — How do I balance the time demands of running this business with my other important priorities? How do I ensure that there is a career path here such that I’m able to continue growing in my skillset and feeling the sense of internal validation and excitement about doing this and be as excited in year six as I was in week six. I think that there is a real focus on sustainability and to building something that lasts there, that the technology industry could use — Stand to adopt that DNA a little more broadly.

Ironically, for a community that’s heavily focused on family and community and the more important things in life, our choice of watering hole is Las Vegas, because who doesn’t think family, community, permanence, righteousness when they hear Las Vegas

[0:46:34.1] JM: There are places you can go in Las Vegas that will convincingly give you that illusion.

[0:46:39.3] PM: Yup, I’m sure there are some folks in Las Vegas who hit all of those things. But I’d say it’s a difficult thing to find when one is walking down the strip.

[0:46:45.5] JM: I agree with that. I know we got to wind down. Has working on Atlas made you critical of any aspects of how corporations in the United States are regulated and patched?

[0:46:58.2] PM: My general feeling with regards to regulation in taxation is that like many small business founders, the numbers don't matter so much. The exact specifics of the regulations don't matter so much. I just want the amount of time that is required to be rather low. I want the amount of certainty with regards to interactions with government to be rather high, and I want the path forward for interaction to be very, very clear.

There are some regularity regimes that hit those things and those are generally positive regulatory regimes to operate under. There are just regulatory regimes that the amount of interaction directly with the government is rather high. The amount of predictability with regards to those interactions is rather low and it feels like one has to spend more of one's time working in the system than having the system work generically for one's self.

I think there are also some emergent trends in the government that are a good thing. I swear, every time a paper form is replaced with web application, which is happening at pace in recent years. That's almost variably a good thing for humanity. It speeds up processing times in most cases. It makes it much less likely that random errors will get thrown in the process when someone fumble fingers, a thing when they're entering data into the computer and just generally plays better, like, "Why would you stand in line for 30 minutes when you can just file a form at 2 am in the morning or whenever you happen to be working on somethings that just have that be done. So that's kind of nice.

Random example. I still have the LCC that I have in Nevada, which used to be my consulting company/info products company, and it has a filing obligation Nevada for use tax every year. I got an email — Rather, snail mail from Nevada saying I was delinquent on my filing for use tax. I slapped my forehead and opened up the URL on the email and said, "Okay. The amount of use tax I owe you for last year is zero. Enter." They were like, "All right. You're done." I'm like, "Great!" No need to call an accountant and figure out what the process is. No need to call Nevada and say, "Hey, can you please international mail me a copy of the form so that I can fill out and then send it to you and then get an answer in six weeks on whether it was accepted or not." One and done, and more one and done interaction with the government I think is a more positive thing.

[0:49:01.7] JM: Oh my God! I know. This is something I don't run a very complex business, but I still get tons of pieces of mail where it's like a piece of mail from some division of a state organization regarding my business, and I opened it up and it's like a lot of texts and it takes me a while to figure out what they're trying to tell me and it all sounds scary and it seems like I owe a lot of money and I'm not sure what's going on. And then I read it like four more times. I'm like, "Okay. I understand. There's nothing to do and I don't owe any money." The number of times per year when that happens, it's like, "I'm wasting a lot of time," like reading these physical pieces of mail that are hard to parse.

[0:49:45.1] PM: I think this is one of the opportunities for a company like Atlas where clearly for the things that are experienced by substantially every entrepreneur, we can build technology that would sort of like intercept the need for that mail in some cases. For example, a thing that you have to do for virtually every state if you're incorporated is to keep a registered agent available. A registered agent is basically a specialized DNS record that says, "In case of lawsuit, file the papers for the lawsuit for this person who will be available at any hour during the business day that's in the state."

So since most businesses don't get sued every year, all they need to do is to maintain their contract with registered agents and make sure they pay up. But that's a 30-minute tax on every company in the economy every year. We're able to save thousands of hours for Atlas customers just by saying, "Okay. We will find a registered agent who will provide the services if required," and then we'll put a little notification on your dashboard if it's time for you to pay them, and then we'll charge the credit card that's normally on file for that and you don't have to get interrupted with, "Hey, it's March again. Time to dig out the number and then call them and swap some paperwork, yadi-yadi-yada." We will just take care of that for you.

[0:50:55.6] JM: Yeah, and other things — Some other things that come to mind are like notaries, or paying franchise tax, for example. Yeah, there's a wide variety of things that, "Please, dear God. Take these things away from me if you can. Atlas"

[0:51:11.2] PM: We have a tool within Atlas to assist users in calculating the Delaware's franchise tax. There's actually two ways to calculate it. Typically, one of the ways is vastly more to the advantage of the company than the other way, and that's like sort of one of the things that

you have to know when running a company, because we can have software calculate both of them in parallel. We can assist users on making the choice that is more beneficial for them, then kind of like protect your limited amount of entrepreneurial bandwidth.

When you figure out over the next 10 years what your company is going to accomplish, we'll probably not accomplish a fundamental reevaluation of how Delaware expects to get franchise tax submitted to it. You're probably going to create value in some other fashion. So you should spend the vast majority of your brain cycles and to creating into value rather than working the mechanics of Delaware franchise tax. We can take that off your plate.

[0:52:00.7] JM: Whenever I talk to people about the startups that are exciting to me, Stripe is literally at the top of my list. And a large part of the reason for that is it seems to me like Stripe has an opportunity to basically be the next Amazon, and I think the AWS of Stripe is going to be Atlas, because it's so easy to imagine levering up this basic set of company establishment features into a wider marketplace of offerings that are low margin, potentially high margin overtime, relationships with networks of service providers, much like AWS has, a marketplace much like AWS has. Maybe that feels far off today, but it seems like a fairly obvious trajectory to move toward for Atlas. What's the biggest vision for what you think Atlas could become?

[0:53:03.7] PM: Biggest vision for I think what Atlas could become? I'll tell you something that Patrick Allison mentioned to me and attempted to convince me to join Stripe. Immediately after he said the sentence, I thought, "Okay. If we execute correctly on this, then we will make entrepreneurship so much easier than it is in the status quo, then you should be able to see the impact of us in global macroeconomic indicators, because entrepreneurship is so entwined with producing growth, not just in the United States but in countries worldwide."

Ultimately, we are unlocking a great deal of potential on the world that is not presently exposed to the best circumstances to achieving its full potential. I think that Amazon is a wonderful, wonderful business and has produced a lot of value for their customers and for the world that generally we have — It's still very, very early days for us. We do execute on something. It's wonderful thing to hear that some of our customers love what we do and that some people see us as a light in the industry. That's great. But I think we have a great deal of humility internally.

One of our company values is we haven't won yet. We are keenly aware of the fact we need to get much, much better at what we are doing presently and we probably still have not yet created the things that will be most impactful over the next 10, 20, 100 years. I do think that the company is thinking over time scales like that. We're definitely not a quarter to quarter bench company, but there are quarterly goals just like there are anywhere. But we're looking to make an impact on the world over the course of time that it will take to generally make a big sustained impact in the challenges that matter.

Having said that, we will not be able to make the size of impact that we hope to make and just having people like myself working here. For those of you who are looking for a job, Stripe is hiring across a wide variety of poisitions both in engineering and elsewhere. So please take a look at stripe.com/jobs, or drop me an email. I would be happy to chat about potential opportunities.

[0:54:56.3] JM: Patrick McKenzie, thanks for coming back on Software Engineering Daily. It's been great to have you.

[0:54:59.9] PM: It's been great to chat, and if I can ever help anyone out, my email address is patio11@stripe.com, please feel free to reach out to me at any time. You can never waste my time.:

[END OF INTERVIEW]

[0:55:12.5] JM: GoCD is a continuous delivery tool created by ThoughtWorks. It's open source and free to use, and GoCD has all the features you need for continuous delivery. Model your deployment pipelines without installing any plug-ins. Use the value stream map to visualize your end-to-end workflow, and if you use Kubernetes, GoCD is a natural fit to add continuous delivery to your project.

With GoCD running on Kubernetes, you define your build workflow and let GoCD provision and scale your infrastructure on-the-fly. GoCD agents use Kubernetes to scale as needed. Check out gocd.org/sedaily and learn about how you can get started. GoCD was built with the learnings of the ThoughtWorks engineering team who have talked about building the product in

previous episodes of Software Engineering Daily, and it's great to see the continued progress on GoCD with the new Kubernetes integrations. You can check it out for yourself at gocd.org/sedaily.

Thank you so much to ThoughtWorks for being a longtime sponsor of Software Engineering Daily. We are proud to have ThoughtWorks and GoCD as sponsors of the show.

[END]