The Distributed Data Show Podcast Episode #137:

Marie Kondo your data with Heidi Waterhouse - TRANSCRIPT

Release Date: February 25, 2020

Heidi Waterhouse: Because we have a lot of feelings about throwing things away if they might be useful. You know what we really need to come.

David Gilardi: Is this some Marie Kondo stuff we're talking about here?

Heidi Waterhouse: Yeah it is. We need to Marie Kondo the shit out of our data.

David Gilardi: From DataStax, this is the Distributed Data Show podcast.

David Gilardi: Hello everybody, and welcome to another episode of the Distributed Data Show. I'm Dave Gilardi here with Heidi Waterhouse. Did I get that right?

Heidi Waterhouse: Yep.

David Gilardi: All right. Heidi Waterhouse from LaunchDarkly and this is actually kind of funny. She's literally standing like right in front of me. We're here at Data Day Texas and I'm looking at my Twitter feed and I happen to follow Heidi and I see her picture show up on my Twitter and then I look up and I look down and look up. I'm like, "are you Heidi?" You know, and then ensued this conversation. So Heidi, could you tell everybody who you are, what you do, and we'll get into it from there.

Heidi Waterhouse: Yeah, it's the hair. You all can't tell on the podcast but I have bright pink hair.

David Gilardi: It is very nice. Is it pink? Mauve?

Heidi Waterhouse: It's fuchsia.

David Gilardi: Fuchsia, fuchsia's the color? Yes, yes.

Heidi Waterhouse: But it makes it easy to find me in a crowd and it's useful to, to stand out. So, I work for a company called LaunchDarkly. We do feature flags as a service which allows you to control your code after deployment. And I'm here to talk about the death of data. I was invited to reprise the talk I gave last year in Austin about how all of our machine learning is predicated on datasets that we have not done a good job cleaning or tossing things out of.

David Gilardi: Okay.

Heidi Waterhouse: We just assume that all data is good, but if you collected my employment information eight years ago-

David Gilardi: Yeah.

Heidi Waterhouse: I have a different job title. I am now a developer advocate and not a tech writer. I have a different salary range. I have a different address. All of that data that you think you know something about me is wrong and yet we're feeding it into machine learning and then it gives machine learning hallucinations.

David Gilardi: So then with, that's pretty funny. So then what is in your talk, do you present a solution to this or guidelines or where's it go?

Heidi Waterhouse: I have some solutions. I say the first thing that you need to do is automate deletion of old data and decide what it is you need to keep and what is unnecessary. You do not need last year's log data.

David Gilardi: Oh, okay. I see.

Heidi Waterhouse: You do not need meeting notifications. You do not need a financial information passed when you are fiduciary requirement to retain it.

David Gilardi: Okay.

Heidi Waterhouse: So please don't.

David Gilardi: Okay. Now is there any idea there to say aggregate, collate or anything, any of the data or no, just cut it and you don't need it? Like, move on.

Heidi Waterhouse: Well I think you should automate the deletion because we have a lot of feelings about throwing things away if they might be useful.

Heidi Waterhouse: Like this is, this talk actually started after I read The Life-Changing Magic of Tidying Up and I was like, you know what, we really need to come.

David Gilardi: Is this some Marie Kondo stuff we're talking about here?

Heidi Waterhouse: We need Marie Kondo the shit out of our data.

David Gilardi: That's actually my wife and I, we went through, now we didn't go full on Marie Kondo. I will say, I realize it was a fad, I realized it was honestly-

Heidi Waterhouse: It's a useful metric.

David Gilardi: It is. And we went through a process like that with very many things. I actually now travel with my, as an advocate and I'm sure you know this, we travel a lot and I Marie Kondo'd, the way I travel and that kind of deal and it honestly there is something to be said about the freshness or the newness of cleaning out things and learning how to better organize stuff and moving on, get rid of it and move on.

Heidi Waterhouse: And so, it's really important to me that as we talk about machine learning and AI, we understand that the data sets we're consuming changes how we feel about things and changes the decisions we make.

David Gilardi: Interesting.

Heidi Waterhouse: And if we are using old data, it's kind of a poison.

David Gilardi: Interesting. Interesting.

Heidi Waterhouse: And the algorithms will be wrong. And so I say that people should automate deletion. I say they should as a company, hire an archivist. So that's librarians. But for companies, there are a lot of people with library degrees.

David Gilardi: Right? Right.

Heidi Waterhouse: And they have specific training in what they call deacquisition.

David Gilardi: Okay.

Heidi Waterhouse: Which is throwing things away.

David Gilardi: Oh, interesting.

Heidi Waterhouse: So they can say this is historical trend data, we need to keep that. And they can also say, this is irrelevant data and we should toss it.

David Gilardi: Okay. So you've got to give him the power, you've got to give them the ability to go ahead and go through with this stuff and kind of get out of.

Heidi Waterhouse: Right or at least market and say, is there a reason we can't throw this away?

Heidi Waterhouse: I've put it all in the pile. We're looking at all the same type of thing. Is it really useful to have this for more than three years? So you know, hire somebody who's a specialist in this, collect mindfully. I think it's really easy for us to collect a ton of data about our users.

David Gilardi: We'll go ahead and we'll minus later and figure it out later.

Heidi Waterhouse: Right.

David Gilardi: Or something.

Heidi Waterhouse: The problem is that's all threat surface. The more data you collect, the more threat surface you have for when, not if, you are eventually compromised.

David Gilardi: I see. I see.

Heidi Waterhouse: Because.

David Gilardi: It's a good point.

Heidi Waterhouse: Everybody who has data has this vulnerability and the more we've collected, no matter how we secure it, we have to give our people access to it and as soon as someone has access to it, it's possible for bad actors to have it.

David Gilardi: Right and we have more data. Right. More of it. Now is there, is there an angle of cost of ownership as well? Cause I'm assuming that if you're, if you're cleaning out your data over time then you're decreasing the footprint that you need. Is it not.

Heidi Waterhouse: Well, you're not decreasing it so much as not letting it explode exponentially.

David Gilardi: Okay. I see.

Heidi Waterhouse: But if you have Gmail,` you never throw mail away because there's no reason. If you have Outlook and you're constrained to like two gigabytes or something.

David Gilardi: You keep it.

Heidi Waterhouse: Well, no, you don't necessarily keep it clean, but you are forced every once in a while to be like, "Oh I got to go through and throw stuff away

because I got the note that says like IT is mad at me, but storage is so cheap right now that we don't have that forcing function on our data set.

David Gilardi: I see, I see. Yeah.

Heidi Waterhouse: And and we keep them all live and I'm like, if you're going to retain that data, treat it like a Bitcoin wallet and store it offline. It's like just don't store it connected because, why? Unless you're actively accessing it.

David Gilardi: Right. And that way if you store it in an unconnected fashion, you now remove that threat. You know, if you do have a breach or something like.

Heidi Waterhouse: Exactly. And I think a lot of people will find if they store something unconnected, they don't access it.

David Gilardi: Okay. Yeah.

Heidi Waterhouse: It's like we can't tell that we're not accessing it when it's connected, but if we have to like "I have to file a ticket to get that drive reconnected" and nobody files a ticket for a year.

David Gilardi: Are you really using that data at all?

Heidi Waterhouse: Then you're not really using that data.

David Gilardi: You're not really using that data.

Heidi Waterhouse: It's easier to see that when it's this disconnected.

David Gilardi: Yeah. That's interesting. So do you, since you gave your talk last year, right? It was something similar?

Heidi Waterhouse: Well, at The Lead Developer Austin.

David Gilardi: Okay. Okay.

Heidi Waterhouse: Then invited me to give it here.

David Gilardi: Okay. Okay. So have you since then, or have you seen any examples of people who have started to embrace this, these kind of guidelines or anything like that or anyone that you know who does this on a regular basis?

Heidi Waterhouse: Yeah, so I have seen some impacts. Mostly what happens is people will watch the talk and they will go back and talk to their governance board

or because mostly bigger companies, right? And they're like, okay, we're going to set up some policies around this. And so I don't know if I can name the companies?

David Gilardi: Yeah, probably not. Yeah, it's okay.

Heidi Waterhouse: But I've seen people like go back and say, "Hey, as part of the data management team, I think it's important that we do this."

Heidi Waterhouse: And then people are like, "Oh yeah, I hadn't thought about that", because storage is cheap and it doesn't hurt and they hadn't thought about the threat surface or they thought about it, but only in a securing it way, not in reducing the vulnerability.

David Gilardi: Okay. Okay. All right. So then, we were talking a little bit earlier as well and you again from LaunchDarkly, you guys work on feature flags, right?

David Gilardi: So is there any crossover between the talk that you're giving with regarding to like archiving your data and cleaning it up and then what you do at LaunchDarkly or are those kinds of two separate things?

Heidi Waterhouse: There's not a ton. Some of it is, I think that feature flags are really interesting for decomposing monoliths.

David Gilardi: Okay.

Heidi Waterhouse: So you just have this like giant ball of mud and you think you've written a microservice that replaces part of it, but you're not sure.

Heidi Waterhouse: Like you're not 100% sure you've got all the bits, so you wrap that bit of the monolith in a feature flag and turn it off.

David Gilardi: So you're almost like soft breaking it up in a way.

Heidi Waterhouse: right?

David Gilardi: Right.

Heidi Waterhouse: Yeah. And so, and you've turned it off, you haven't actually pulled any code out.

David Gilardi: Okay.

Heidi Waterhouse: And you see if it breaks.

David Gilardi: Ahh, that's neat. Okay.

Heidi Waterhouse: Yeah. So like most people when they're thinking about feature flags, they're thinking about turning it on and doing like canaries and progressive deployment and roll outs and giving people a lot more safety and turning things on. But I think that it also gives us a lot of safety and turning things off and down without necessarily doing anything destructive.

David Gilardi: Okay.

Heidi Waterhouse: And so I think for a data hygiene perspective, I think we could also be doing that. Like if I flag all of this data off, if I flag this API, this stream, this database off and rerun the query, what happens?

Heidi Waterhouse: But you haven't, you haven't destroyed anything. It's, it's nondestructive testing.

David Gilardi: Right? Right.

Heidi Waterhouse: And you're like, wow, that's a really different outcome. Okay. Now I can dig into why it's different and whether I need that old data and it's giving me some useful historical perspective or I'm using historical perspective that is no longer correct.

David Gilardi: Right. Okay. Now with that, do you guys from the LaunchDarkly perspective, are you advocating for folks to use feature flags or you know, and presenting them with guidelines and stuff with you have applications and tools and things that they can use to do this. How does that part work?

Heidi Waterhouse: So, what we have is about almost 20 SDKs now.

David Gilardi: Oh wow. Okay. Is this stuff that you could embed in your application and then no way.

Heidi Waterhouse: Yeah.

David Gilardi: Oh, okay. Let's hear it.

Heidi Waterhouse: Yeah. Okay. So what happened is,

David Gilardi: I didn't know this.

Heidi Waterhouse: Yeah. You as a developer wrap the piece of code that you want to control with a little call that says if-then-else essentially it's the fancy if flag, like all feature flags are fancy if statements.

David Gilardi: That's right. That's right.

Heidi Waterhouse: All machine learning is also a fancy if statement, I'm not wrong.

David Gilardi: I have to think about that one for a moment, you know?

Heidi Waterhouse: Yeah. So you say you write that in the code and then that gets picked up by the SDK and presented to an API.

David Gilardi: Okay.

Heidi Waterhouse: You can either control the API directly or we have a beautiful, shiny console that lets people do targeting and feature flag management in a gooey based way, which is super important because I was just at Delivery Conf, which was awesome in Seattle and I realized that the problem that people are having with continuous integration and continuous deployment is that they're not separating deployment from release. Deployment is a software operations problem.

Heidi Waterhouse: Like getting stuff on the server, whatever server means in this case that's, that's a software problem. Release is a business problem.

David Gilardi: Okay. Yep. Right.

Heidi Waterhouse: And we should not be making developers make business choices and we shouldn't make business people beg developers for.

David Gilardi: That's a fair distinction, definitely. Okay.

Heidi Waterhouse: And so I think it's really exciting to be able to talk about splitting up that area of control and what does feature flagging mean for how we do operations? What does it mean for alerting? What does it mean for pipelines? What does it mean for personalization? LaunchDarkly has this architecture where we have most of the logic on the client side. Wherever you're serving from ,whatever client is. The fact that we can have a client with a client with,

David Gilardi: Right, right. Turtles all the way down.

Heidi Waterhouse: It is client turtles all the way down, but wherever your page is serving, wherever your activity is taking place.

David Gilardi: Right, okay.

Heidi Waterhouse: The logic exists there and so it doesn't have to query back to some master flag database and say, "Hey, what's my state, it knows what state is. And that just continues to hum along happily until you push a server side event that says, "Hey, I want you to evaluate these rules differently." And because it's just a flip of the flag, it's already been deployed. It's like you've deployed Schrodinger's code, it's on and off at the same time and then you're going to flip the flag and say, I want to deliver this now or I want to deliver this to 10% of Spanish speaking people in Florida.

David Gilardi: Oh, so you have controls like that?

Heidi Waterhouse: Yeah.

Heidi Waterhouse: Everybody we talked to already has that data about their users. Like if it's a logged in user, you know where they live.

David Gilardi: Do they really from a code perspective, even if they have feature flags, how many times do you come across an application or a team that has already implemented each flag that have that level of fidelity where you can control it quite like that. It's usually on or off, right?

Heidi Waterhouse: It's usually on or off because it's super hard to do that level of targeting. It's expensive to do that. Everybody could be doing it. They have that data, you know there's a database of your users that says we know these things about each user profile, right? You have that. There's also development knows you could, I suppose specifically target IPS that you had associated with user profiles, but that's hard, which is why we have a whole company doing it.

David Gilardi: If you put that on every development team to try to figure that out, right. That's not going to have, it's probably not going to have an equal amount of effectiveness. Effective deployment, if you will. Right. So it sounds like it's not just feature flags but feature flags with sharistics and then the controls to allow you to kind of gain where you're going to put the flags.

Heidi Waterhouse: And we just added a whole experimentation set which allows you to say, okay, I'm going to push out, AB, okay, I'm going to do AB testing. And when one of these is statistically significantly winning, I want you to flip everybody to that one.

David Gilardi: Interesting. Honestly, you need to talk to Cedric, shameless plug for my colleague Cedric Lunven. I hope I got your name right again, Cedric, because it's a joke with us.

David Gilardi: But anyway, you know we were talking earlier that Cedric was one of the creators of FF4J, right. You know, that's a feature flag system and such. And we've talked about this before and coming from my particular history and previous companies. I've done the type of feature flagging that given our conversation would be very primitive compared to what you're talking about, right? But it's like it, it sounds like I'm actually like, this is something I just want to bring him to the stuff we're doing because there is a level of sophistication here that would have been really nice for the things that I was doing in the past to have that there and be able to control that stuff.

David Gilardi: So, I guess it goes back to those SDKs. I'm assuming that you have all those SDKs cause you have one per language kind of thing and that you embed that in your code and then you can essentially use that through an API.

Heidi Waterhouse: That's right. I'm, well, I'm not showing you, I'm remembering. Oh I, I never remember all of them. And this isn't all of them, but we have Java C++.net, Python, JavaScript, Ruby, go PHP, iOS.

David Gilardi: You're covering the bases. You're covering the main bases.

Heidi Waterhouse: And we've gotten to the point where we're like now having debates about; are we going to do COBOL, are we going to do Hass?

David Gilardi: Oh wow, you're going for COBOL?

Heidi Waterhouse: Well, okay, we've gotten asked for COBOL and we're like, COBOL doesn't think like that. You know. So I don't know if we'll actually end up doing it, but it's at least a discussion point because although the delivering to end users feature flag story is the easiest one, there's also some really interesting ops stories. Like for instance you're, you're a grumpy IT admin right? And you're managing a giant database that has APIs writing to it all the time, right? This is a common scenario. If one of your write-in API's goes nuts and starts writing in a hundred X, its usual rate, what do you do?

David Gilardi: What do you do? Well, at that point maybe you're shutting down an application instance or whatever.

Heidi Waterhouse: So, you might try and take down the server, you might try and disconnect it,

David Gilardi: But that causes disruptions, all sorts of things.

Heidi Waterhouse: So, what I'm saying you can do is you can put a circuit breaker pattern, feature flag on that inbound pipe and connect it to your metrics. So, if you say, if I see a 10 X spike in rights, it's probably a bad action. Shut it off and notify me so I can figure out what's going on before it writes all of this garbage to my data.

David Gilardi: So how does that apply to things like, let's say that you have some online presence or like an eCommerce site or something like that, but maybe you have something or capability where that just goes completely viral. How do you distinguish between, Okay, maybe 10 X or a hundred X and it's blown up and hopefully your system can handle that. Hopefully you're sized appropriately, but you know how do you distinguish between that are essentially good, right, and then the ones that are essentially detrimental.

Heidi Waterhouse: Yeah, so I think it sort of depends on what your threat matrix is. If you think there's any chance of like virality you say alert me, but don't shut it off.

David Gilardi: Interesting.Okay.

Heidi Waterhouse: Or I think another interesting one is throttling. You don't have to do an absolute shut off. You could do a couple things.

David Gilardi: It's pulled back a little bit, make sure you don't bring your system to its knees kind of thing.

Heidi Waterhouse: Right. Exactly. So like Best Buy went down a few years ago on black Monday, cyber Monday, and they couldn't stand back up. Not because they didn't have the compute to handle the load, but they didn't have the compute to handle the startup load because everybody was reloading the page at the same time because they came back and they get all of this heavy page load all at once.

David Gilardi: Throttling capability from that standpoint, from the client, it could have been at least a slowed down some of the event.

Heidi Waterhouse: Or you can have it a page set up so you're like, please serve a degraded version but don't fail.

David Gilardi: I see.

Heidi Waterhouse: So, I'm not going to serve animations. I'm not going to serve ads. I'm not going to serve a tractors. I am going to serve the core business value.

David Gilardi: Okay. Right.

Heidi Waterhouse: And so if you're like, "Something has happened, we've gotten", I keep saying slash dot dot dot, I'm sorry the youth, but it was this a thing where your site would be humming along with a hundred readers a day and then all of a sudden you'd get a referral and you get 100,000.

David Gilardi: Right. Right.

Heidi Waterhouse: And it would just crush your computer cause we didn't have scalable cloud.

David Gilardi: That's right.

Heidi Waterhouse: So, if that happens, you would rather serve your core business value then go down altogether.

David Gilardi: That's right. Yeah. That makes sense. So, you've got the SDKs, you take that, you embed that in your app just like you would any other SDK. Then through that, I'm assuming you're going to use whatever methods and such that are exposed in there.

Heidi Waterhouse: Yep.

David Gilardi: Essentially, the handle would have been the if-then-else conditions.

Heidi Waterhouse: Right.

David Gilardi: We would have done with the hard coded or not hard-coded but like the configured feature flags. Right. You know, cause I remember back in the day you'd have a configuration file, you have all the feature flags in the file.

Heidi Waterhouse: Right.

David Gilardi: And then if you had, you could flip them and hopefully you didn't have to restart your app or-

Heidi Waterhouse: push the file.

David Gilardi: But here you're saying that once you enable this right now, you essentially have like a, if you will, that has an interface you can kind of toggle everything and do it real time.

Heidi Waterhouse: Yes.

David Gilardi: Okay.

Heidi Waterhouse: And by real time we're talking under half a second for the US .

David Gilardi: Okay.

Heidi Waterhouse: So like once it's deployed,

David Gilardi: You flipped the switch and now it will propagate out throughout all of your clients and such.

Heidi Waterhouse: Worldwide, we just added streamers in Asia and Europe so that we're,

David Gilardi: Oh, okay. So this is going to be a secondary question then, what is the backend? What is handling the communications? It's something that the clients supply on their own, or is that something that LaunchDarkly does.

Heidi Waterhouse: Launchdarkly does it.

David Gilardi: I see.

Heidi Waterhouse: We have a network of streamers and CDNs. And.

David Gilardi: So when you're using your SDKs, and you're looking stuff up, right, there's got to be a part of it that's talking back to a home base somewhere.

Heidi Waterhouse: Right.

David Gilardi: Okay. I get it.

Heidi Waterhouse: Yeah.

David Gilardi: Okay.

Heidi Waterhouse: So when you start your application, it instantiates, it bootstraps itself and it says, okay, what's my flag state? Who am I? Where am I at?

David Gilardi: Okay.

Heidi Waterhouse: And that's the heaviest part. But for most applications, there's a lot of load that happens right when you start up and instantiation so it doesn't add that much. And then after that it can evaluate on those rules and it's almost no, no ways.

Heidi Waterhouse: Does that answer the question?

David Gilardi: Totally answers it. All right, well thank you, Heidi.

Heidi Waterhouse: Yeah, thank you.

David Gilardi: This was very enlightening and in like a totally random conference conversation that came out of nowhere.

Heidi Waterhouse: Twitter.

David Gilardi: Yeah. There we go. This is where Twitter works. Twitter works just now.

Heidi Waterhouse: Just now we'd like to know the miracle.

David Gilardi: And now I know more about LaunchDarkly and other people and more about LaunchDarkly. And of course, your talk at DataDay.

Heidi Waterhouse: Yes. Thank you.

David Gilardi: What's your Twitter handle if you want to get that out?

Heidi Waterhouse: My Twitter handle is @wiredferret. I know.

David Gilardi: That's awesome.

Heidi Waterhouse: We pick these things out in 2007, we did not know.

David Gilardi: Wow.

Heidi Waterhouse: Yeah.

David Gilardi: Oh, that's wonderful. Well, mine is Sonic DMG.

Heidi Waterhouse: Yeah.

David Gilardi: I've had the name Sonic since I was a teenager, so.

Heidi Waterhouse: Exactly. Wired was a long time ago.

David Gilardi: So, wiredferret. Well, thank you Heidi. I appreciate you coming on and thank everybody for listening to another episode of the Distributed Data Show. See ya.

Heidi Waterhouse: Bye.

David Gilardi: Looking forward to seeing you at DataStax Accelerate, where we'll provide a whole set of community talks selected by community leaders for everyone and the open source Cassandra world. And don't forget to register in either San Diego or London by going to www.datastax.com/accelerate.