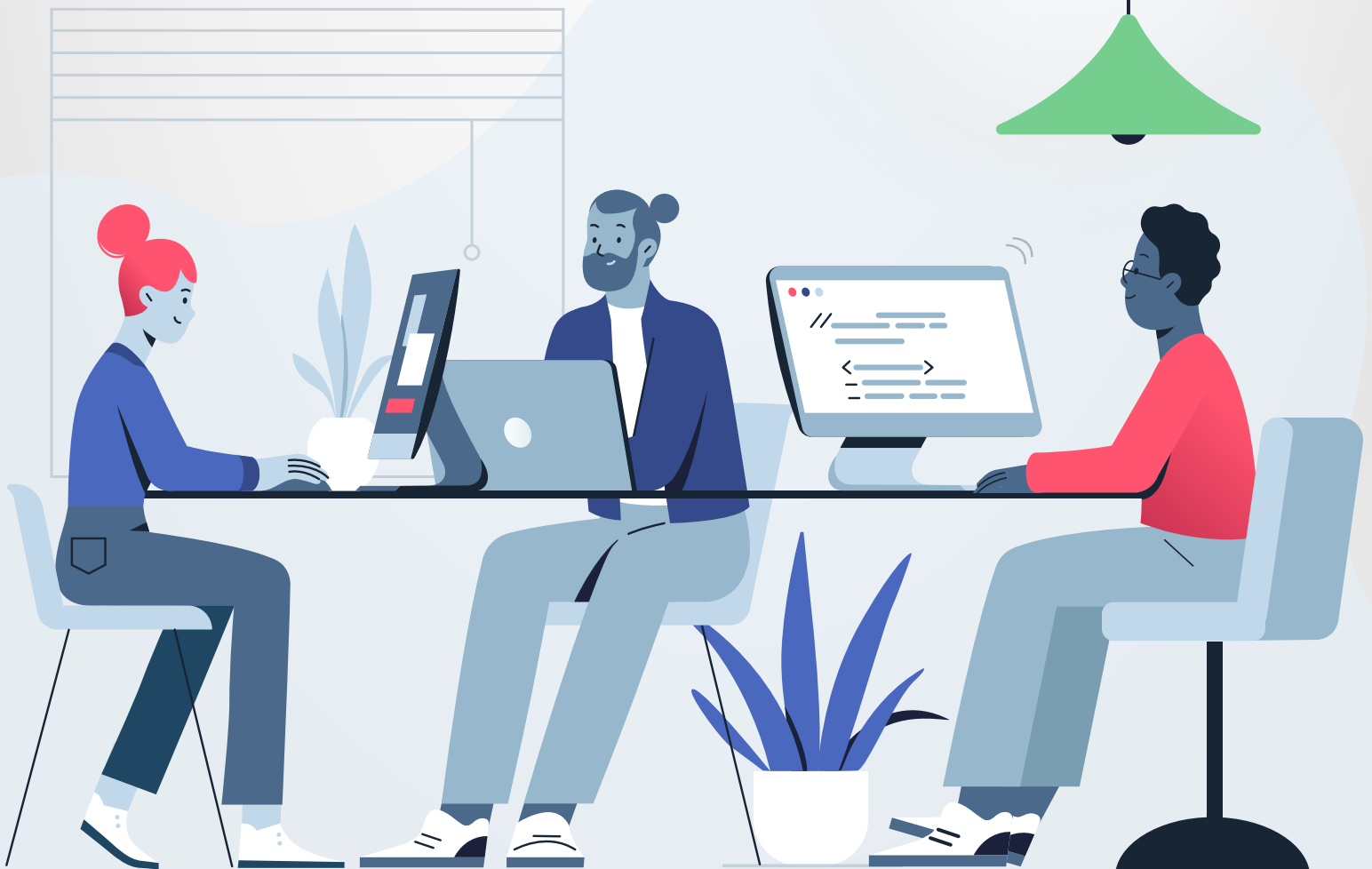


# The Observability Odyssey

Days in the Lives of DevOps, SREs and IT Ops

**Helen Beal**



# Table of Contents

■	<b>PROLOGUE</b> .....	3
■	<b>CHAPTER ONE</b> .....	5
	Sarah Tackles Anomaly Detection Using Intelligent Observability	
■	<b>CHAPTER TWO</b> .....	10
	James Reduces Noise and Distraction and Fixes a Network Issue with Hybrid Cloud	
■	<b>CHAPTER THREE</b> .....	15
	Dinesh Fixes an Issue with a Legacy MQ System and Eliminates Some Toil	
■	<b>CHAPTER FOUR</b> .....	20
	Sarah Looks Under the Hood of AIOps with the Cloud Team	
■	<b>CHAPTER FIVE</b> .....	26
	James and Dinesh Petition the CIO & CFO for an Enterprise Rollout of AIOps	
■	<b>CHAPTER SIX</b> .....	33
	Dinesh Presents a Talk with DataDog at AICon OKR	
■	<b>CHAPTER SEVEN</b> .....	38
	Sarah Experiments with Observable Low-Code	
■	<b>CHAPTER EIGHT</b> .....	43
	James Embarks on a Service Desk Migration to Improve Incident Management with AIOps	
■	<b>CHAPTER NINE</b> .....	49
	Dinesh Experiments with Chaos Engineering	
■	<b>CHAPTER TEN</b> .....	55
	Sarah Resigns from Animapanions and Heads Off to a Competitor	
■	<b>CHAPTER ELEVEN</b> .....	61
	James Speaks with the Industry Analysts	
■	<b>CHAPTER TWELVE</b> .....	67
	Dinesh Starts an AI Community of Practice	
■	<b>ABOUT THE AUTHOR</b> .....	73
■	<b>RESOURCES</b> .....	74



# Prologue

**ON BOXING DAY IN 1836**, Bridget Murphy decided it was time she moved to Dublin. As soon as she could, she set out from Galway, with nothing but the clothes on her back, some bread and cheese in a bag and a picture of her family. And her needlework skills. Taking a post on her arrival in Ballsbridge as a dressmaker, she married the love of her life, Philippe Bergmans, a wool trader in Ghent in March 1837.

Their twin boys arrived on September 4 – Connor and John. The boys had not shone in the schooling their parents invested in, being much more interested in football and money. At fifteen, in 1852, they purloined a broken cart from their father, fixed it, filled it with groceries bought from the cash they'd stolen from the biscuit tin on the dresser, and rolled it out on the cobbled streets in search of customers. C&Js was born!

One cart became two and then ten as the brothers hired the start of their workforce. Then came a shop, then seven shops, then a march across the borders as they expanded into France, Germany, Luxembourg, and the Netherlands. They acquired a soap factory. And a brewery. And invested in fruit farms.

By 1865, C&Js was a recognized brand across the European continent, with flagship stores in London, Dublin, and Paris, and employed over twenty thousand people. It had acquired six of its closest competitors and made the family extremely rich. John died in a sailing accident on his yacht in 1880.

Connor continued to search for ways to make more money and, on his deathbed, encouraged his two eldest sons to take the boat to Houston, Texas in 1915 to cash in on the oil boom. They broke into America. They had inherited their father's and uncle's hardworking and entrepreneurial skills and ran the company for the next fifty years until they retired to the golf course. They left the company in the hands of the CEO, Donald Reinertsen, who they poached from a competitor, impressed by his innovations in the credit card and shopping trolley.

Donald had an eye for innovation and in 1974, was one of the first to introduce bar codes. Warehousing, inventory, and ERP systems soon followed and thus C&Js had an IT department. By 1990, Donald long-retired, they had over one thousand employees building and maintaining their IT systems across the company who reported to the IT Director, who later became the CIO. Growth had stalled C&Js agility and they were no longer the innovator they once were. They didn't jump onto the eCommerce bandwagon until 2005, so distracted by M&A they were, and have been playing catch up on digital transformation since. Lack of investment and treating the IT department as a cost center, meant that decades of technical debt built up, silos were formed and when Charlie, a new CIO arrived in 2018, he doubted he'd stick it out beyond two years and wondered what change he could really make.

His first real win was to bring an acquisition to the board. Animapans was a startup he'd been watching (and using extensively for its subscription services to keep the family pooch, Leonard, in biscuits). It was right in C&Js wheelhouse - they already had a huge share of the pet food market through their stores. Charlie saw them as a lean startup model he could encourage the rest of the business to observe and learn from. They were killing it with DevOps and disrupting the marketplace. Bringing them and the team into the C&Js was a massive win for him and his plan to drag C&Js into the newest, 21st-century ways of working.



## CHAPTER 1

# Sarah Tackles Anomaly Detection Using Intelligent Observability



**T**HURSDAY MORNING, and I've done some yoga, a ten-minute meditation and am at my desk in my hastily thrown up garden office with a mug of green tea by 08:30am. I'm really not missing the commute to our old HQ (now permanently closed, thanks to the pandemic) in the heart of Seattle and am enjoying an extra few minutes in bed and getting mindful before logging in.

I start by checking Slack. The company I work for, Animapanions, is an online pet store. We were acquired by a global conglomerate headquartered in Texas a year back. We are continuing to run most of our own systems, but our warehousing and delivery has moved to be controlled by our owning company. We process several thousand orders a day, shipping a few millions of dollars of product each month.

When I logged off last night, all was looking good but, since the takeover, we've started shipping internationally and run 24/7 so anything can happen while I'm catching my zzz's. While I'm recharging, our teams in Manila (where the owning company's central NOC is) take over following that sun.



I see an invite to a new Slack channel and catch my breath as this often means we've got a problem. Everything's been strangely quiet since a new AI-driven observability platform was put in place a couple of weeks ago. It's been great getting to some of the improvement backlog that was aging rapidly, and I'd been starting to wonder if we'd previously spent an unnecessary amount of time chasing our tails, but also worrying it was some kind of calm before a storm....

I sip my tea and dive in. The new Slack channel is called "Moogsoft Incidents." There's a message from the IT Ops guy aligned to our product team, based out of the Texas HQ:



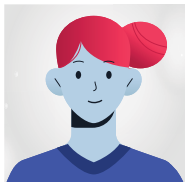
**James Parker** 08:02 AM

An anomaly has been detected in the Animapanions customer journey – checkout. **@Sarah Edwards**, please can you take a look and let me know what you think? The app's still up atm...

I'm the Sarah that James is tagging, and there's a link to an incident which has created a Jira ticket. First, I want to see who else is in this channel. I'm a DevOps Engineer and I've been on the team since we created the original app. I was involved in the shopping basket and checkout dev so it makes sense that James is asking me to look into this. The problem is that we've always operated with a "we build it, we own it" mentality, so it puts me slightly on edge to have someone I barely know effectively looking over my shoulder. I grudgingly thank him in my mind though, for alerting me to what might be a problem we need to deal with.

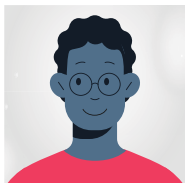
Also in the channel is a support engineer from the Manila NOC and support center, Lamar Ramos. But nobody else from my team. I bring Dinesh, our SRE, into the Slack channel too and make sure he's got access to the Jira ticket. He's new to the team and doesn't know the code like I do, but he's a skilled programmer and he'll want to know about this as it could be impacting our SLOs. Plus, it was James and he who implemented the AIOps tool.

I take a look at the incident, digging out from the depths of my memory the short demo of this tool James and Dinesh did for the team a month or so ago. What I see in there makes me sit up straight and my brain starts to whirr.



**Sarah Edwards** 08:23 AM

Thanks **@James Parker** – I've taken a look and **@Dinesh Soni** is also here. What I think I can see is that Moogsoft's found an anomaly on the new Samsung Galaxy S21 11.0 Android browser that released on Monday...



**James Parker** 08:25 AM

Yep – it looks like a few customers haven't been able to complete checkout – any clue what's going on?



**Sarah Edwards** 08:27 AM

I'll spin up a test in AppliTools. **@Dinesh Soni** – you want to jump on a screenshare?



What I'd seen when linking to the incident was a number of alerts coming in from New Relic, some Splunk log files and some data from Prometheus. We undoubtedly had a problem, but where? Dinesh and I did some triage with AppliTools but couldn't find anything in the UI, so we went back to the Moogsoft incident. Moogsoft showed us that the transactions were breaking at handoff to PayPal – but it looked ok to ApplePay. Was PayPal down? It couldn't be, because the transactions were working just fine in the other browsers....

Back in the Slack channel, I queried our CI server to see what changes we'd pushed into live yesterday. There were a couple of small features from two of our US based developers, Priti and Sean. I invited them into the channel too.

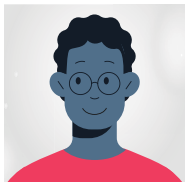
It turned out that Sean's change had included an API update which had broken the connection to PayPal in the new version of the Android browser used by that particular device. He made a quick code change and pushed it through the CICD pipeline while we all watched the progress through the Slack-based ChatOps platform. I brewed another cup of tea while we waited for the tests to

pass and he manually pushed it to live. The Moogsoft alert switched off and we saw the transactions going back to normal.



**Sarah Edwards** 09:38 AM

And we're back! **@Sean Davis** you're a superstar. Thanks so much for fixing that so fast. Quick retro everyone? Then we'll attach this conversation record to the Jira ticket Moogsoft created.



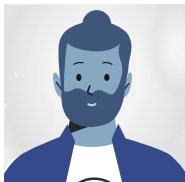
**James Parker** 09:39 AM

Amaze. I'm pretty happy with how that went.



**Sean Davis** 09:41 AM

Me too. Sorry for the issue guys – not sure I could have anticipated it but I'll work with the testers, probably Ling, to see if we can add some automated integration tests to the CICD pipeline to pick these up. I have to say, I feel like we would have been thrashing about in the dark trying to figure out what was going on without that intel from Moogsoft. Thanks for setting that up guys.



**Dinesh Soni** 09:45 AM

Yep. I'll second that. Looks like the Moogsoft experiment's working in line with our hypothesis. Time to take it to the next level, **@James Parker**?

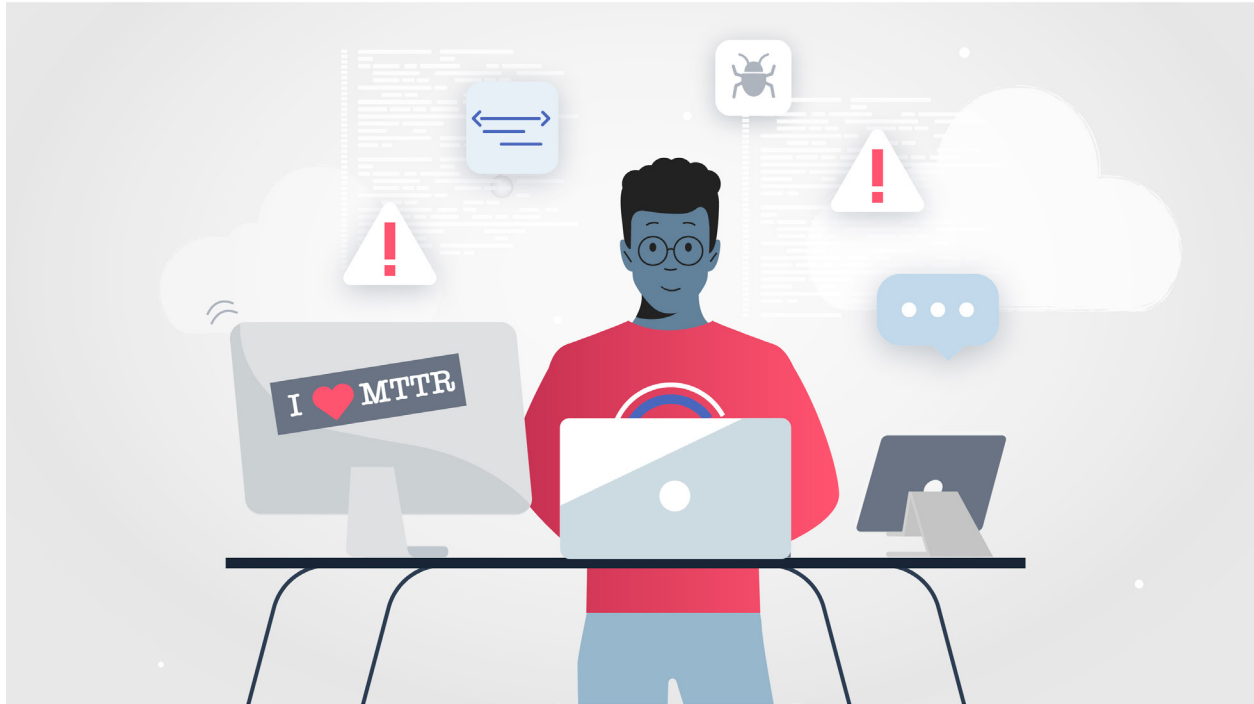
## CHAPTER 2

# Dinesh Fixes a Legacy MQ System Issue and Eliminates Toil



**A** **NOTHER DAY**, another SEV 1. Something feels different today though.... It's not just that it's my twenty-year anniversary at C&Js, although that in itself is a bit of an achievement, if I do say so myself. Some of my mates laugh at me for still working at the same company I joined fresh out of college with a mechanical engineering degree, but, as I tell them, it's never been boring. C&Js is one of those vast global conglomerates nobody really knows about, but half the stuff in your kitchen and bathroom cupboards is made by us. A couple of brothers, Connor and John, started a grocery cart in Belgium in the mid-1800's and things kind of snowballed from there. When I joined, as a noob on the support desk, the internet revolution was just starting, people were still giving Amazon a hard time for not turning a profit.

Speaking of Amazon, I've actually come into the office to see for myself the impact of our cloud migration. There's hardly anyone here of course, social distancing, masks on, hand sanitizer everywhere and all that. Our data center is not yet visibly shrinking but I'm here to decommission the first machine.



I'd made it all the way to lunchtime without any unplanned work - a miracle in itself. In the old days I'd've hoped to be taken out for a Texan barbeque or something fun in Austin, where we are in the HQ. I still miss home, England, and our world-beating pubs and warm beer, but they're not open anyway. So my celebration lunch is a SEV 1 and a pizza I called in.

So, the SEV 1. It's been a roller-coaster ride during my two decades - most recently I've been working with the cloud infrastructure team we formed about six months ago. Aparna leads it - she's been here nearly as long as me - she joined me on the support desk back in the day. It was all AS400s then. We still have a few of them but they're all virtualized. She's a bit of a force, Aparna. Pushy, some say. I admire her for her vision and tenacity. The respect is mutual - we still always make sure our teams are connected and we get to work together. She got an AWS architect certification on her own time, then petitioned the CTO to set up a team and then hired four new cloud engineers. She wants to cross-train the department but they can never find the time to learn - never the time to save time. They're too busy sprinting to stay still; trying to change the wheels of the car when driving is tough.

BUT! She has just completed the migration of her first application into AWS. The machine I'm here to decommission is the one it used to run on. It's an inventory management application that one of our smaller subsidiaries uses. She's set up a Zoom call so I join.



On the screen is a sea of alerts. As usual I also have a ton of emails from our different tools. My own screens are going crazy but I can't see what's going on. It's like looking for a needle in a haystack. Alert fatigue is a real thing here. I can feel my eyes glazing over and my brain going numb.

"Hey, James, thanks for joining us," Aparna greets me as my video pops up.

"No problem," I say, giving her a wave. "Awesome, you're here too, Dinesh," I greet our SRE happily - this is a bit of a dream team moment we've got going on here. Aparna's whole cloud crew is on the call too. "What's going on?" I ask.

"It looks like transactions are failing from the inventory app to the on-prem database," says Aparna.



“Pushed any changes through, lately?” I ask, tongue-in-cheek. It’s a bit like asking a user to turn their machine off and on again. The tech equivalent of “take an aspirin.” Always start with the basics.

“Nope,” says Aparna. “We’re giving it some time to bed in. Maybe there’s a problem with the database.... I’ve got so many alerts from the monitoring systems - there’s so much noise I really can’t see the wood from the trees. I can’t believe we’ve broken it already. This does not bode well for our next experiment.”

“Dinesh, did you...?”

“I did, James. I did.” Dinesh is grinning and answered my question before I had a chance to finish asking it. Last week, we had an incident with one of our more recent acquisitions, Animapanions, and got to road test our new toy - our AIOps and observability tool, Moogsoft. We resolved the incident in record time and got back to something more interesting instead. After that, Dinesh had said he was going to set it up for this app for the next experiment. So glad he’d managed to get to it.

“Can I take over the screen?” Dinesh asked Aparna and then threw up the Moogsoft view of the incident. It correlated alerts from the database, the CICD toolchain, the cloud app and infrastructure. And the network.

“Interesting,” I said. I could see the problem.

“What is this? What am I seeing here?” asked Aparna.

Dinesh explained: “This is some AI looking for patterns in the alert streams we’re getting.” Then he started singing, “I can see clearly now—”

“It’s not us, is it? It’s the network. I can see it too - that spike on the CPU and that Kubernetes container and all the network resets. Wow - that’s amazing. Talk about noise reduction - it’s cut right to the chase. Let’s get the NOC on the call.”

In less than five minutes, they were there too. Ten minutes after that, the network problem was resolved. Two minutes after that, we had another problem - another massive CPU spike as we hit the 'thundering herd' condition when all the failed transactions retried and succeeded. Aparna managed that via AWS and our database server took it - then the alert noise faded, and the system purred again. Our work here was done. I saluted the team and left the Zoom, hearing the familiar clack-clack of Slack as the screen closed.



**Dinesh Soni** 12:34 PM

That was pretty awesome fun, mate! You got a minute to look at something else with me? There's something up with MQ...

## CHAPTER 3

# Dinesh Fixes an Legacy MQ System Issue and Eliminates Some Toil



**AFTER WE'D FIXED** Aparna's network issue, James came to see me at my desk. Masks on, socially distanced and all that, but it was nice to have some face-to-face time. James is cool - that dry British humor and not your classic IT Ops dude. He's been here forever and mentored me when the CIO, Charlie, hired me as the first SRE here a year or so ago. I lucked out really. It was good working at my previous company, and great that I was able to upskill from sys admin to SRE, but, frankly they were a bit of a mess. I wasn't really able to do my job properly as nobody ever made the time to save time. There was too much pressure from the business to get new features out and never space to work on antifragility. They were old school and I was worried that C&Js was going to be the same - they have been around for over a hundred years after all. But Charlie seemed to have the right vision for their digital transformation and the package was good so....

"What's up with the old MQ?" James asked as he sat down.

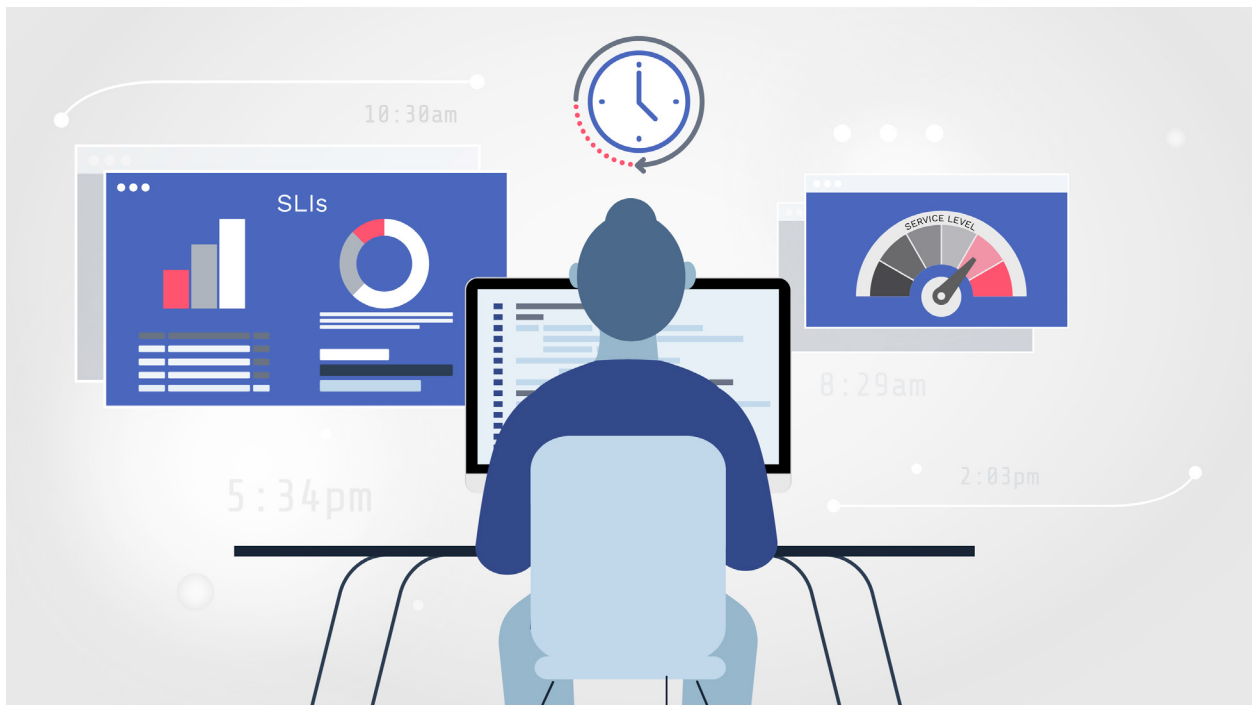
"These legacy systems—"

"Cherished," corrected James with a wry smile.

“Take a look at this,” I said, pointing at the screen with Moogsoft displayed on it.

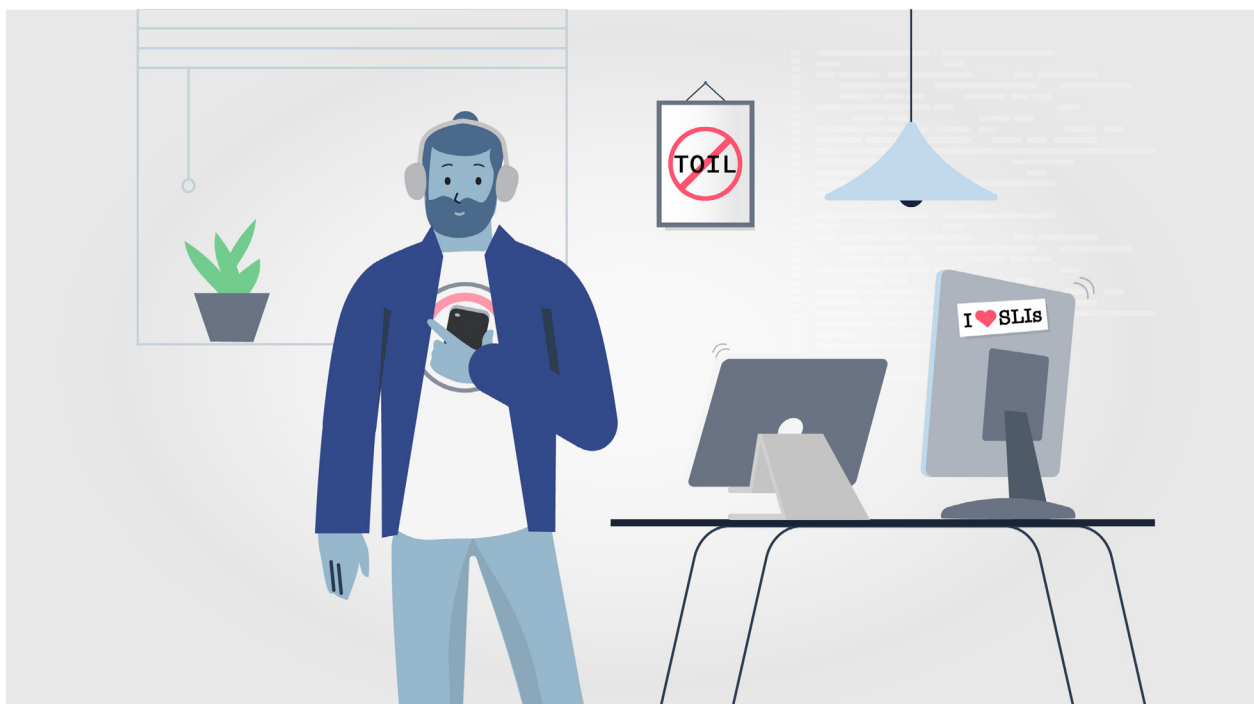
When I said that James isn’t your typical IT Ops guy, this is an example of what I meant. The IT Ops team I worked with before, they were so busy with unplanned work, hair-on-fire moments, we were just lurching from one disaster to another. Almost exclusively caused by the huge amount of changes development was pushing through to us. Not only was the quality of the code not great, the technical debt was enormous and building every time a new change came through. To be fair to the guys, I think they were just burned out. The staff turnover was massive and new people would arrive bright and bushy-tailed but, give it a couple of months, and they’d look exhausted and frustrated too.

It’s different here. There’s still massive amounts of technical debt, don’t get me wrong, but these guys can see it. And they’ve got control over the work coming through - the development teams are working with the IT Ops teams to visualize the flow. And James doesn’t get in a flap when things go wrong - he’s really calm. And he fights to invest in things that’ll make things happen faster. That’s why he sponsored me to the CIO to pilot Moogsoft.



While we were on that call with Aparna, I saw another alert relating to the message queuing - an ancient but pretty bulletproof to-be-fair tech that connects our inventory and eCommerce systems. Actually, it connects most of our systems. The monoliths anyway. I was chatting with Jorge last week - he's the chief architect here. He was telling me about his master plan to break down the monoliths using the strangler pattern. He thinks he can break all these dependencies. Goodbye CABs and hello team autonomy. Hallelujah! In the meantime though, we still need to manage this.

So Moogsoft has this thing where it uses time series components, things like periodicity models, and data seasonality. You know, like Friday nights are a spike for Netflix. And Black Friday's really busy for retailers. But there's quite a lot of subtlety to it and this is why we often get so many alerts that don't mean anything - there are natural spikes and waves in performance but most monitoring systems aren't clever enough to get it. Moogsoft compares a sequence of events that's occurring to a sequence that happened in the past that was bad. This pattern analysis gives us an early warning, pre-empting failure if you like. It's not prediction - that's impossible - and if we were able to predict, that would show us we weren't doing a very good job since, if we could predict it, we should have already fixed it.



“The dynamic threshold’s being breached,” said James.

“Yeah,” I said. “I think we’re about to have a wobble.”

I’d barely finished my sentence when my other screen, which displayed a dashboard I’d created to bring together the multitude of alerting tools that have been collected over the years, went bonkers.

“I’ll just do a restart,” I hit the keyboard and got on with it. One of the things that is great about MQ is that it has assured delivery. You can do things like this and not worry that you’ve lost any transactions. It remembers where it got to and will start where it left off. The restart worked.

“Ok, that’s great. It’s getting a little flaky in its old age,” said James.

“Yeah. Jorge’s thinking of using a new cloud or open source MQ platform in his new microservices empire,” I said.

“Good call,” And this is another example of how awesome James is. In my old place, the IT Ops guy would have walked away at this point. Fire out, job done. But James isn’t done yet. “You had to do that a lot lately?”

“A couple of times last week.”

“And we don’t know what’s causing it?”

“I have a couple of theories but it’s going to take me a bit longer to figure it out. I was getting close to the end of my experiment just then and then it fell over so I’ll need to start the test again. In the meantime, I’m thinking we could automate a fix. Restarting is boring toil.”

“That is the rule of three, after all,” James said. “Done it three times, time to automate. What are you thinking of doing? Restart or redeploy or raise the threshold?”

“Automated restart. Sounds a little scary but we’re getting this early warning so I should know when it’s about to happen. It only takes the system down, and it’s going down anyway, for a few minutes

and it'll catch up where it left off. The impact on the users should be minimal."

"Good call," James nodded. And then took things another step further. "What do you think about making MQ a candidate for cloud migration with Aparna? Maybe we could get Jorge involved too and see if we can try switching some out for a newer alternative. Another little experiment."

"I like it," I said, grinning under my mask. I could have fixed this on my own, but it turns out sharing and collaborating has all sorts of benefits.

## CHAPTER 4

# Sarah Looks Under the Hood of AIOps with the Cloud Team



**B**ACK-TO BACK ZOOM CALLS as usual today, but this next one I'm pretty excited about. It's three weeks now since that incident, the API change that broke the Paypal connection in Android, and I haven't seen a lot of our SRE, Dinesh. I have to share him with a few other teams and one of them, Aparna's cloud team, has been keeping him really busy with a legacy migration into AWS. Next up though, is an AI masterclass I asked him for as, while he's been largely absent, he hasn't actually been needed by my team all that much, but I want an opportunity to pick his brains.

Since he and James put the Moogsoft pilot in, our MTTR has been close to zero and consequently, we've been able to use all that time not spent on unplanned work on paying down technical debt and reducing toil ourselves like Dinesh suggested. Our product owner is over the moon as we're now at a point where we've managed to inject more capacity for innovation - we're taking on more story points per sprint now than ever before. And I've had time to tinker about with integrating Moogsoft into the DevOps toolchain, which is also partly what this next session is about. And because I'm curious to get under the hood of this AI and understand how it's doing what it's doing.



“Hey Sarah,” Dinesh greets me, his happy, smiling face on screen and ready to go as soon as login. We chatted for a few minutes while waiting for Aparna and the rest of her crew to finish whatever call they just had and dribble into the Zoom. I know some people are complaining about Zoom fatigue, but really I’m just super grateful that I get to work from home and not commute and that it’s such a gift to be able to get a bunch of amazing people together in an instant no matter where they are in the world.

“James and I have a meeting with Charlie next week. We’re pitching him for an enterprise-wide roll out of Moogsoft so this is a bit of a dry run for me today - thanks for asking me to set it up.” Dinesh looks pretty excited about his plan to chat with our CIO.

“Wow,” I responded. “That’s a pretty big deal!”

“I know, right,” he says with his lopsided smile. “It’s fifteen countries, over five hundred teams, and close to ten thousand people. It’s not just the software license fees, but all the time and effort. Do you think you can hit me up with a success story?”

“Of course! Let me post some links in the chat. Here’s the internal wiki page where I wrote up that first incident you and James sorted with us. And here’s an article I posted on our public-facing engineering blog last week. Also, here’s a link to my experiment documentation for integration into the DevOps toolchain.”

“That is so awesome,” he says, beaming. “We’ll be making your local discoveries global improvements in no time! After this, let’s set up some time for just the two of us as I want to know more about the DevOps toolchain plan. Looks like everyone’s here now, so let’s get going.”

Dinesh starts by doing some intros and I learn some new things about him straight off - like he's studying for a data science Ph.D. on his own time. To add to his existing computer science degree and neuroscience Masters.

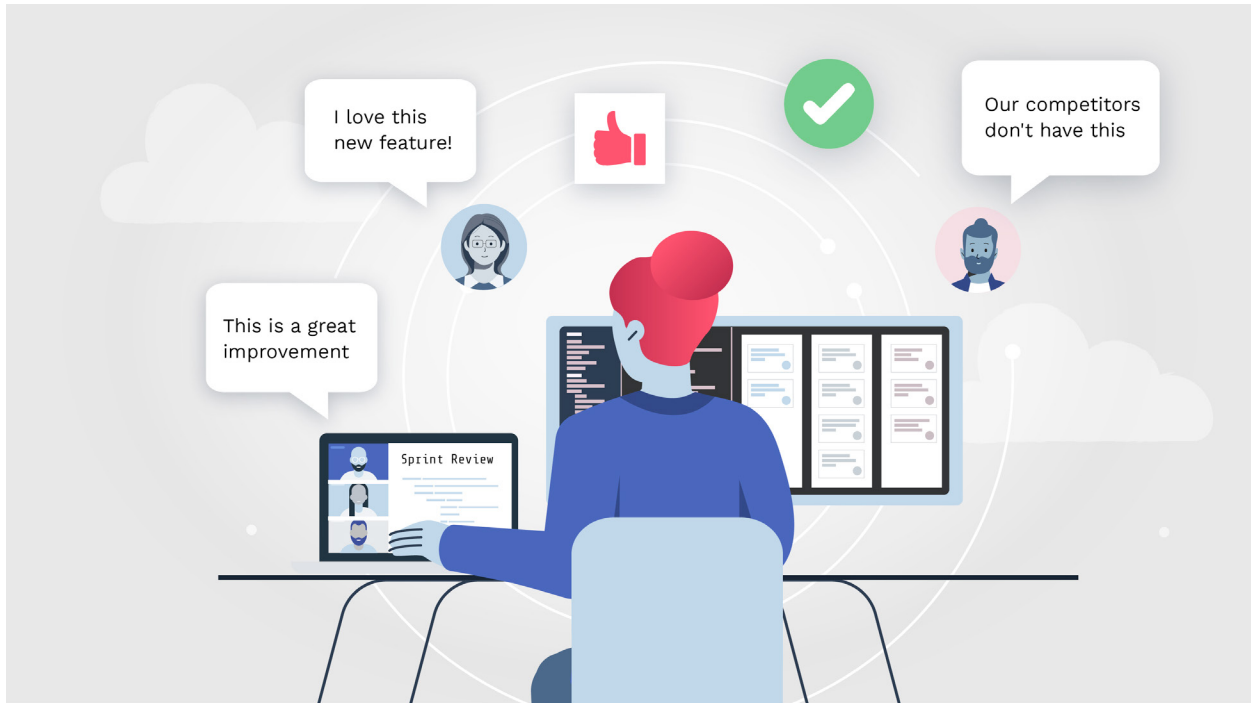
“So, enough about me,” he says, “welcome to the first of what I hope will be many of my AI masterclasses! I’m really grateful for you wanting to do this with me, thank you, as I love showing off.” We all laugh, with him. “Seriously though, this is a really fun way for me to self-assess how well I’m learning and I’m super geeky about this so I’m hoping I’ll get you as excited about AI as I am and then I’ll have more people to talk with about it.”

His enthusiasm is infectious and I’m even wondering if I could carve some time out to do a Ph.D. too. Data scientists are highly sought after and highly remunerated after all.

“The great thing about Moogsoft is you totally don’t have to be a data scientist to benefit from all the work the data scientists have already done for you. But it’s fun to know what’s going on under the bonnet. Today, I thought we’d focus on noise reduction. As I’m sure you’ve all experienced, sometimes having so many alerting systems is a real problem - there can be so many false positives and false negatives it can be really hard to get to the crux of what’s going on. Moogsoft reduces this noise by automatically applying statistical calculations and noise-reduction algorithms to that bountiful alert data. And there’s our first AI buzzword: algorithm.”

“Sounds complicated,” says Aparna.

“It does, right?” agrees Dinesh. “But it’s not - it’s just a set of instructions. The algorithms that are reducing noise do quite a few different things. First off, there’s deduplication. That’s nothing new, it’s been around for years, but it’s simple and effective so we start there. Essentially, every time a repeat event is encountered, a counter is incremented on the parent alert, and the repeated event is discarded. So, hundreds of ping fail events collapse to a single alert. With me so far?”



Nods and thumbs up all around. Looks like everyone's hanging on his every word! I know I am.

"What's next?" I ask.

"Next up, is alert correlation which is all about pattern discovery across technology stacks."

"So network, server, and application?" I asked.

"Yes," confirmed Dinesh. "And in your newer environments and Aparna's cloud native pieces, microservices and containers too. And it's not bamboozled by all these cross-enterprise dependencies we have either."

"It can help us manage the dependencies while we learn to break them," I murmur.

"The correlation algorithms weigh across multiple factors. The Cookbook algorithm, for example, creates clusters of alerts based on how alike or dissimilar they are. It uses factors like time, class or type, geographic location, topology proximity, and server priority."

“So it zips up alerts into batches?” Aparna asked.

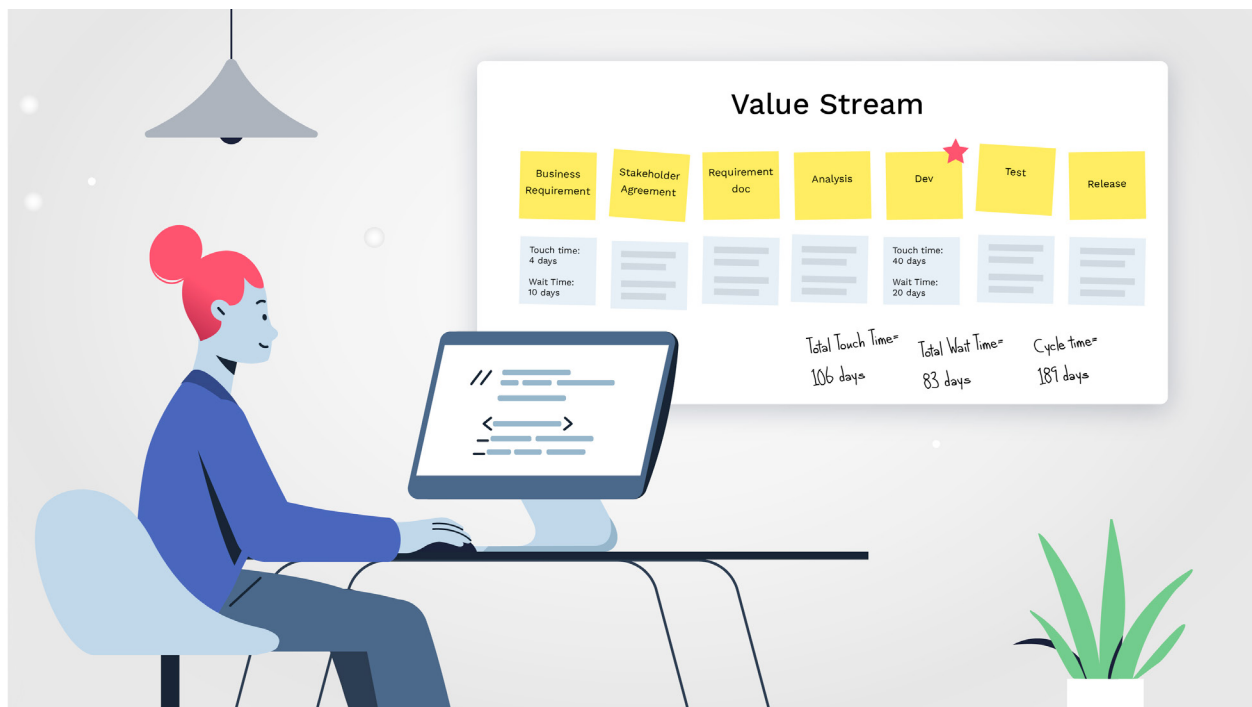
“Yeah, kind of. But as well as vastly reducing the number of tickets you’ll get, the correlations also teach the system to identify where root cause, or causes, are likely to be and make recommendations on how to fix it.”

“So we can automate those fixes!” I’m getting excited. This is sounding like the closest I’ve been able to get to self-healing systems in my career.

“Totally!” said Dinesh, nodding frantically. “And this is also helpful when we get to higher levels of capability and we’re not just trying to fix immediate, urgent issues, but also looking for other improvement opportunities.”

“Imagine that!” says Aparna. “Having time to save time!”

“I know, right,” smiles Dinesh. “But that’s what happens when your MTTR is close to nil, and you’re barely having any change failures or outages. It’s a self-fulfilling prophecy.”



“What else does it do?” I ask, I’m getting seriously hooked on this stuff.

“Check out this visualization,” Dinesh said, sharing his screen. Of course, I’ve seen this before but it’s interesting to see the “wow moment” on their faces that must have been on mine just a couple of weeks back. “And you can fine-tune this too.”

“Oh,” I say, “that’s really useful. When I’m thinking about how to integrate this into the DevOps toolchain, and that we want to shift all testing left and do it as early as possible, I think what you’re saying is that the system will know where it is in the route-to-live and how important that stage or gate is to us. Can we fine tune it if we really want to amp up building that quality early?”

“For sure,” says Dinesh. “Let’s look at that before my meeting with Charlie. See you all at the next masterclass?”

“For sure!” the team echoed.

## CHAPTER 5

# James and Dinesh Petition the CIO & CFO for an Enterprise Rollout of AIOps



“**MORNING, MATE,**” I greeted Dinesh as he walked into the office. “Nice get up for the big day!” He was wearing a pressed shirt, rather than his usual hoodie.

“Thought I’d make an effort, you know,” he grinned.

We’d been planning intensely for this moment for the last week or so for our meeting with Charlie, the CIO, to present the results of our Moogsoft experiments and ask for permission to extend the rollout across the enterprise. We had slides and everything.

Charlie was already in the boardroom when we arrived. It was rare for me to set foot on such hallowed ground; I could probably count the times on one hand despite the 20 years I’d been here. Charlie had company: Lucia, our CFO. The nerves kicked in and I glanced at Dinesh, his forehead unusually shiny. I started praying our numbers added up, which is not my strongest suit.

We took our seats, distanced carefully around the vast, glossy table and I handled the chit-chat while Dinesh wrestled with connecting his laptop to the screen. Once he looked settled, I started.

“We are gathered here today,” I began and was pleased to see smiles wash across Lucia’s and Charlie’s faces – maybe this wasn’t going to be an uphill struggle. “To consider how AIOps can improve our organizational performance.”

I’d been in these situations before, many times, justifying investment in time, energy, and money to senior business people who, quite rightly, didn’t necessarily understand the ins and outs of the technology I was proposing and I’d learned to start at a high level. Some of my bets over the years had been good ones, not all, but enough to keep me here this long. I think it would be fair to say I was a safe pair of hands.

The pace of change and the available tech had exploded in recent years and it was pretty much impossible to keep up and do “the day job.” I’m glad Dinesh was here. While he was relatively new to the organization, I had heard Charlie describe him as “a breath of fresh air.” His presence was calculated to counteract any concerns people might have about me being too institutionalized, or not up with the latest and greatest. Dinesh had been hired on the strength of his leading-edge, up to the minute skills and, although he sometimes expressed frustration at the pace of change and the high levels of legacy systems, ways of working, and technical debt, he was hired as pivotal to our digital transformation. As such, my assessment that we would be considered trusted advisors on this should be well-founded.

“The key problem we are dealing with here is unplanned work. Our plans to accelerate the flow of value to our customers are stymied when our people are under so much pressure to keep the lights on. We believe we know how to reduce the time it takes to deal with these events and incidents, to allow the teams more time and insights into how to tackle the causes of the unplanned work, pay down the technical debt and start building systems that take care of themselves, that will be reliable, robust and antifragile.”

Lucia raised an eyebrow at the last word I used. I made a mental note to rein the jargon in.



“Dinesh has prepared some data for you to review, based on the experiments we have been undertaking with the cloud-native product, Animapanions, the cloud migration team, and some of our legacy systems.” I smiled at him and he sat up just a little straighter as he took the floor and flicked to his first slide on the screen.

“Sarah, the DevOps engineer at Animapanions, has been using Moogsoft for around a month. During that time, they have experienced around twenty incidents —”

“Oof,” said Charlie. “Why so many?”

“They push new code to live on-demand, typically around five times a day. They have automated almost all of their unit and integration testing and have made massive inroads into their user acceptance and security testing too. They have two main problems though that are giving them around a 20% change fail rate. One is that they have to deal with a number of 3rd party systems, including our own since we moved their warehousing and delivery to ours, and some like payment gateways, which they can’t control.”





“Fair enough,” said Charlie, steeping his fingers. “The other?”

“Despite being a relatively young company, born on the web if you like, and using most of the latest and greatest technology, they do still suffer from technical debt.”

“Ha!” said Charlie. “I doubt they know the meaning of the words! Surely that pales into comparison to what we’ve managed to build up over the decades? And surely they should have known better?” He looked at Dinesh.

“It’s pretty typical actually,” he responded. “Market disruptors like Animapanions usually put themselves under a lot of pressure to get those differentiating features out and architectural and code shortcuts are often made. And once they’re done and the increment is released to live, it’s often onto the next new thing that’s going to win them more customers, and those improvements or little fixes often go unattended for quite a while. That’s partly why I and the SRE role exists. You may also have noticed that I didn’t mention performance testing in the context of what they’ve automated in their CICD pipeline.”

Lucia was starting to glaze over just a little bit. We were slipping down the jargon slope again.

“So tell Charlie how you helped Sarah fix it,” I nudged Dinesh.

“Right,” he said. “On this slide, what you can see is Animapanion’s MTTR.”

“That’s Mean Time to Recover, Restore – there are a few Rs,” I said for Lucia’s benefit, hoping she wouldn’t find this patronizing. “Basically, how long it takes us to get back up and running when something breaks.”

“Thank you,” she smiled warmly.

“You can see, this data shows it’s dropped to nearly zero.” He flipped to the next slide. “Here’s the internal wiki page where she wrote up the first issue Moogsoft helped her fix. Essentially a new mobile browser version caused a problem with our API to PayPal which meant that some users weren’t able to check out. Here’s what Sarah said about this: “We would have been there for hours without the Moogsoft AI. It took the data from all of our monitoring systems and made sense of it. It spotlighted the problem, took all the usual guesswork and frustration away, and guided us almost instantly towards the right fix path.” She also wrote a public-facing account of her experiences over the last couple of weeks on our engineering blog.”

Lucia looked perturbed. “We do that?” She asked. “Isn’t it a bit dangerous to wash our dirty linen for everyone to see?”

“I asked the engineering teams to do that,” responded Charlie.

“There are lots of reasons why: motivating them to share success stories, showing our customers transparency, gaining a reputation for technological advancement to inspire our own and attract new talent....”

“It’s had over fifteen thousand views, and nearly two thousand shares on social media. There are thirty-two comments, all

positive,” Dinesh added. “The reduction in MTTR is time straight back for the team – in Sarah’s case, those 20 incidents used to cost the team on average 8 hours work, and the outages themselves would last on average around 20 minutes. So that’s 160 hours saved per month, assuming zero, so let’s say it’s actually reduced to 7 hours work per incident, that’s 140 hours back. A 20-minute outage at peak trading is just shy of \$1,000 per minute to \$20k per incident, so potentially \$400K per month regained through not having the system down.”

“\$4.8 million a year?” Lucia leaned in.

“Yes,” I said, “and that’s not even considering the reputational damage and how that impacts customer loyalty, our net promoter score, reviews, and referrals.”

“So, what happens if we extrapolate that number across the whole of C&Js?” asked Lucia, I could almost see her frontal lobe glowing as she started to make the calculations. Fortunately, we’d already done them for her.

“We have made some assumptions – we’ve taken the same sort of proportional reductions across the business and teams’ capabilities do vary. For example, some have a lower change fail rate, but they also have a much lower deployment frequency – something they’ll be trying to address as part of the digital transformation in order to continue to compete. But here are the headline numbers as we see them.”

“Five hundred teams and ten thousand people, but you’ve said half the teams are platform teams and haven’t associated a transaction regain number with them. What does that mean?” asked Lucia.

“As part of the digital transformation and DevOps journey,” Charlie replied, “we are identifying which teams are value streams, so revenue-generating, and which are supporting them – they’re the platform teams.”

“Oh lord!” said Lucia. “Not another reorganization?” CFOs, in my observations over the years, aren’t a big fan of big-bang transformation and the new organizational designs that typically come with them. The negative impact on productivity can hit the bottom line pretty hard.

“We are aiming for a much more evolutionary and incremental approach. But as you can see by the numbers from Dinesh and James, we’re thinking this improvement could still represent up to \$1.2 billion not lost in transactions and it looks like a further 840 not wasted looking for a needle in a haystack.”

“Those are some big numbers,” Sarah acknowledged.

“And we haven’t even talked about what the teams can do with that time when they’ve reduced their MTTR. Those are just the hard savings. We’ll have happier people, delighted customers so more of them buying more. Our teams will find themselves in a virtuous circle where they can spend the time they’ve saved making more system improvements for customer experience, and the tech can help them find them.”

“Where do I sign?” laughed Lucia. Charlie winked at me.

## CHAPTER 6

# Dinesh Presents a Talk with DataDog at AICon OKR



**W**HEN I ASKED CHARLIE FOR PERMISSION to attend this year's AICon (virtual, natch) I thought it would be a shoo-in; learning's part of my OKRs after all. But he never makes things easy and his "yes" came with a caveat that's typical when dealing with him. This time, he claimed he didn't have the budget for the ticket (a likely story!) and I'd have to find another way to get one.

I wondered if any of our vendors that were sponsoring the event had extra tickets, so I reached out to my pal at DataDog, Morgan, to ask if they had one going spare. They said no, but they did have a speaker slot and if I joined them as a speaker, I'd get a speaker pass.

Now, I don't do a lot of public speaking and, like most people, the thought of it makes my knees knock a little bit. But I really, really wanted to be at this conference, what with my data science Ph.D. on the side. And the keynote was one of my AI heroes, Captain Michael Kanaan - only the AL and ML lead at the U.S. Air Force. In another life, I'd have been a fighter pilot. Top Gun's my favorite film. Anyway.

Those pesky OKRs also had a bit about publicly sharing our experiences. I'd asked Charlie about that when he asked us all to put that in there. I'd also noted that giving us OKRs is hardly in the spirit of autonomy and empowerment - but he said it's a collaborative process and this had come from the very top - the CEO no less! I'm all about sharing, but in my experience, this has mainly been with our colleagues - not washing our dirty laundry in public. And while we talk about sharing success stories, what makes a good story is peril, suspense, and redemption. The kind of "all fine here" image we generally want to show as a listed company does not make a good story.

Charlie said that the CEO had heard that transparency is all the rage. And that being open was good for our image (as long as we resolved any issues in the right way and it wasn't just moaning!) and the HR, sorry, people team were pushing him as it's seen as a great way of attracting new talent to our business. The engineering blog's also been stood up for this purpose.

So I've been working on my presentation skills. I'm pretty nifty with slides so I focused a bit more on my speaking skills. And the whole being on stage thing. But it's not really a stage, it's Zoom and I'm on that all the time so how hard can this be? I've been watching a lot of Toastmasters on YouTube and Morgan and I built a slide deck together. We're on in one minute.

And we're live! Morgan does the intros - they're a product manager at DataDog and we've titled our talk "SuperCanineCowMan: Integrating DataDog and Moogsoft for SRE Heroes." And I'm up. I definitely got an adrenalin buzz but I'm on familiar ground so I feel confident as I set the scene.

"I've been using Datadog for around five years now, in three different roles so I was really happy when it was already in place when I arrived at C&Js. I've been loving the marketplace since they launched it last summer. It's been a great way to find out about amazing tools that integrate with DataDog and it's been exciting watching new additions pop up in there it seems like every day."

I handed over to Morgan here and they gave a potted history of the marketplace and all that. Then: "Today we're going to focus on the Moogsoft integration though and Dinesh wants to share some stories with you about what's been going on at C&Js."



“I sure do!” I said as I took back control of the slides. “Observability is a hot topic for me as an SRE, and Datadog’s insights have got me out of heaps of trouble in the past. But we all want to be faster. It’s uncomfortable triaging incidents when the clock’s ticking and we all hate letting our customers down. I’m interested in anything that can get me closer to guaranteeing a sublime and delightful experience.” I saw myself smirking a bit in the video at that. Possibly the downside to not having to actually stand on an actual stage - having to watch your own face. I do want our customers to be having a lovely time for sure, it’s just sometimes the marketing speak gets a bit much for me.

“Also, I’m really interested in AI. Right, you all get that.” My attempt as a joke. We’re at AICon! We’re all in this together. Morgan chuckles so that helps a bit with the lack of real-time audience feedback.

“We’ve got a ton of monitoring tools at C&Js, not just my favorite, Datadog,” I say with a wink and Morgan chuckles again, encouragingly. “I counted them all last week and we currently have 24 different monitoring tools that I could find - monitoring all



sorts of different things of course, but that's a hella lot of data. Of course I want to aim for zero downtime and zero incidents, I'm an SRE after all, but I also totally support my compadres who want to pump enhancements through the system to get those value outcomes realized by our amazing customer base. And the reality is that things do go wrong.



“Our DevOps experts are doing all they can to catch and fix problems early, and I know they never let a known defect downstream, but this is about unknown unknowns. We also practice limited blast radius techniques like canary release but bad things do happen, even to good people.

“Combining Datadog and Moogsoft together creates SuperCanineCowMan and now I can be an SRE hero. Thanks, Morgan! Using Moogsoft’s algorithms in conjunction with Datadog’s insights automates the identification of critical, actionable data. We also ingest data from some of those other monitoring systems and enrich that too. Then meaningful alerts are correlated into

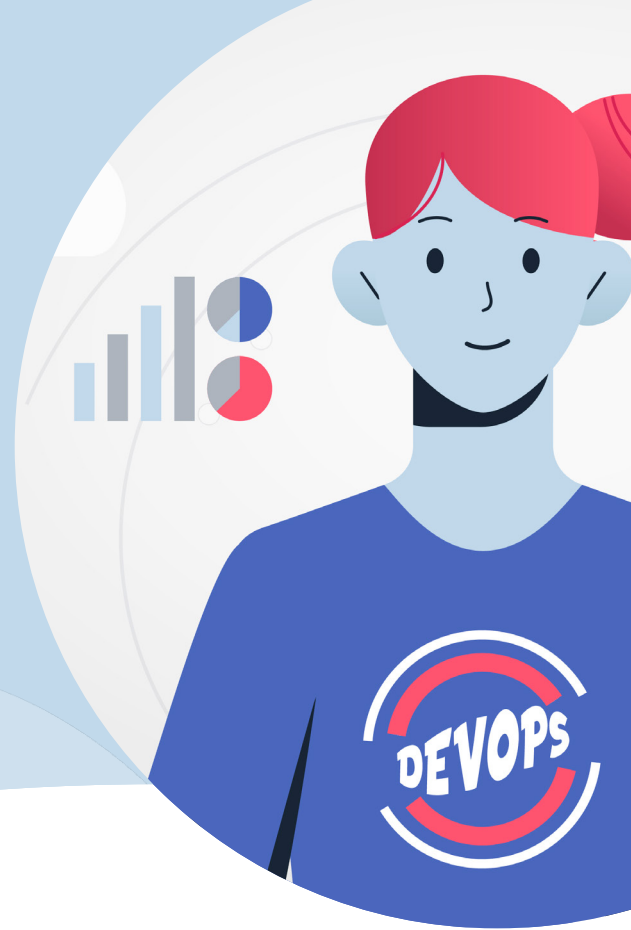


context-rich incidents. I can then quickly resolve them from within the Datadog Incident Dashboard. Let me show you something that happened recently with our cloud migration team.”

I did a live demo and it worked just fine! Morgan had warned me, and I’d seen enough to know they weren’t just being super nervous, that live demos are notorious for going wrong. We had a ton of questions from the audience on the event chat channel straight after and made some new friends. I got some answers to some questions I was struggling with from one of the other speakers and I got to see the keynote at the end of the conference with Captain Kanaan and won a copy of his book, *T-Minus AI*. A bunch of people connected to me on LinkedIn - we’re interviewing one next week for Aparna’s team! All in all, a good day. And I got to check off an OKR. Charlie was pleased.

## CHAPTER 7

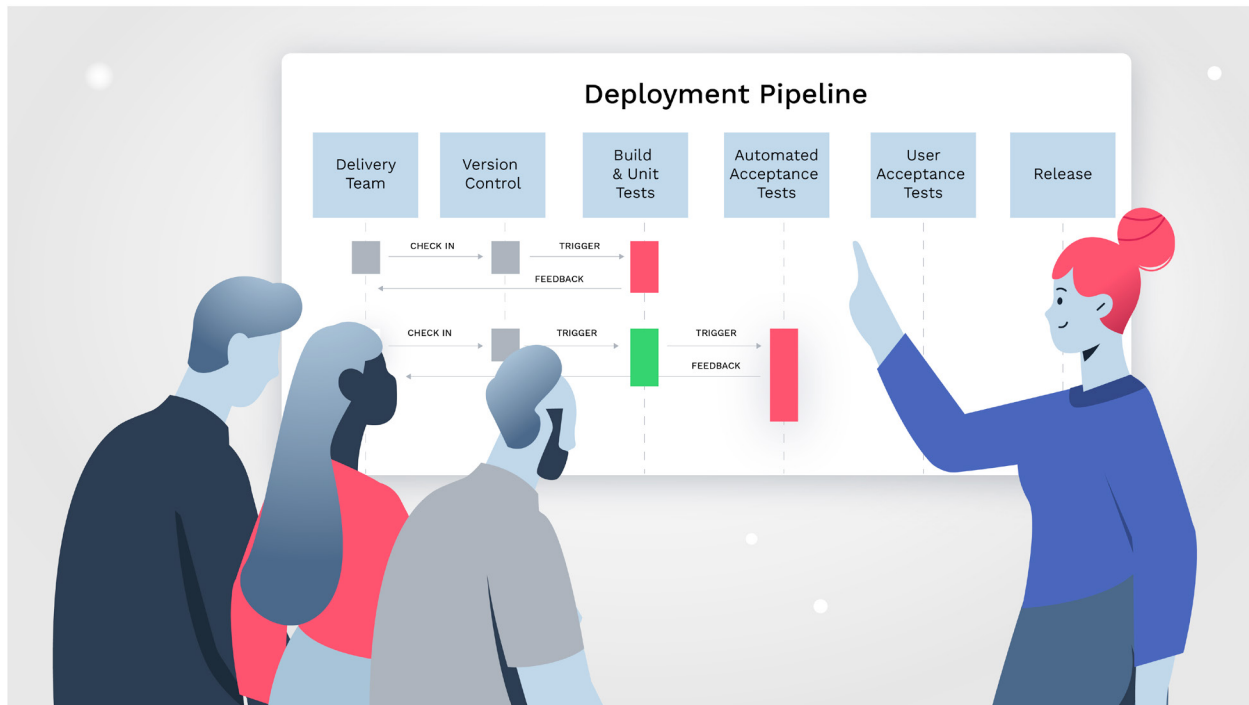
# Sarah Experiments with Observable Low-Code



**I'M KIND OF HAPPY** I became a DevOps engineer, as it increasingly seems that developers are losing their cachet. Well, maybe that's a bit strong, but last time I built a bit of the Animapanions product, I barely wrote any code at all - I composed most of it from open-source binary artifacts from online repositories.

Actually, even as a DevOps engineer, while I write some scripts, a lot of what I need to do is in the tools. But last week, we were in a difficult situation. One of our competitors had released a new feature and their app reviews were going wild. Our product owner, Pat, said we needed to match it forthwith.

The feature was a pet breed wishlist. All human slaves of, for example, Russian blue cats, can form a community in their app and make recommendations based on what their felines really like, or what they thought they'd really want. I get it. I am a slave to PrincessPeachyMelba. She's a Russian blue and it's a constant battle to keep her crazy little mind entertained. I blow a good chunk of my salary on toys for her. Specific breeds do have particular characteristics. What's more, humans get pretty passionate about the breeds they love and cohabit with. And as Pat said - all that customer data! Imagine what we can do with it in terms of product placement, marketing, buying, and cross-selling.



In this morning's stand-up, we broke the rule about making no changes in the sprint and pushed out the tech debt user stories I've been working on until the next one. To be fair, this is the first time this has happened and I'm confident Pat's not going to be doing it again in a hurry.

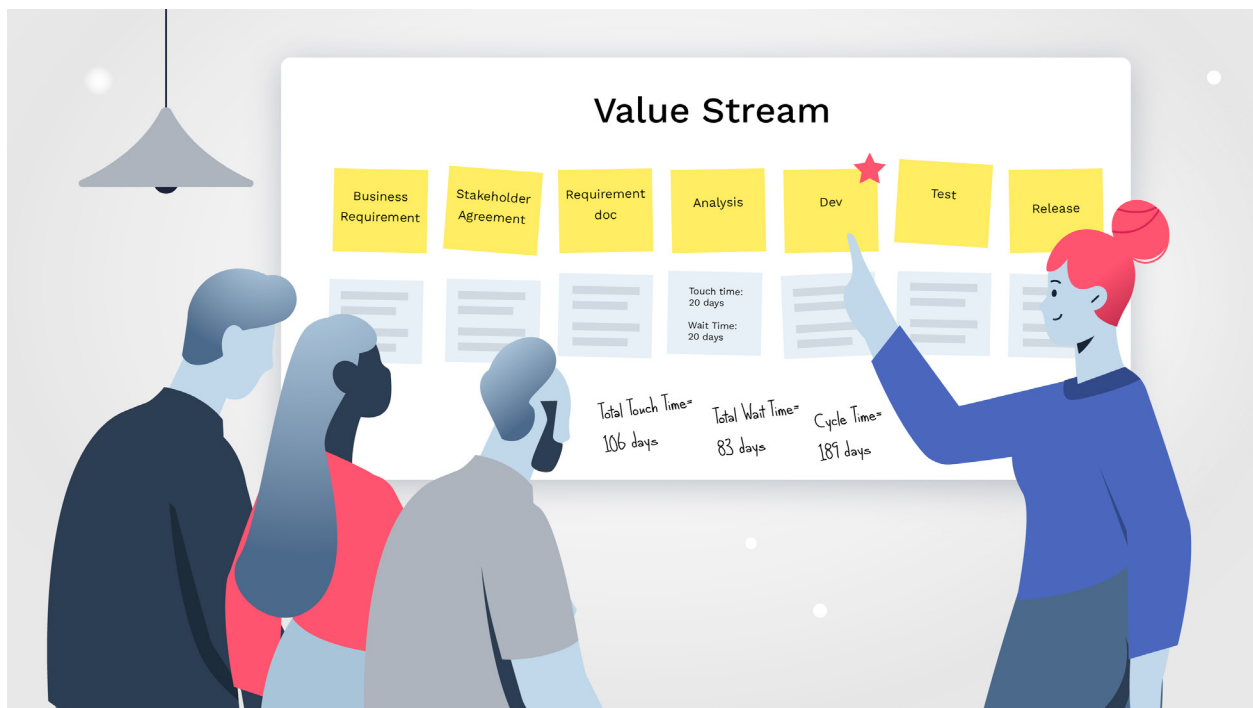
It was time for me to put my developer hat on once more. But when I sat down to code after that standup, Jason, our lead developer rolled his chair over to mine and said:

"You want to try something new?"

I'm all in on experimentation. You might say it's in my blood; both my parents are scientists. My mother's a science teacher, retired now. But I had the rare pleasure of being educated at the school where she taught. Not as bad as you might imagine; she was pretty popular as teachers go. But rarely a lunchtime passed without guffaws from my peers, just out of her class, at yet another failed experiment. I understand now that it doesn't matter so much if experiments fail. The important thing is that they have a hypothesis, are empirical and we respond to the feedback. In our case, that means developing a small new feature, testing it out with a small number of users (we practice feature toggling, A/B testing, and canary releases extensively), and comparing

the results to our original hypothesis to decide whether to pivot or persevere. It's the classic plan, do, check, act - also known as the Deming cycle.

It makes me wonder what our competitors had hypothesized for their new feature and what success criteria they had defined in terms of value outcomes. We can only see what they are getting from their app reviews and only imagine what it's done for their market share, new customer count, basket size, and all the other things that positively correlate with sublime customer experience and make for a high-performing organization.



“You’re on,” I replied to Jason. “What’s cooking?”

“A couple of weeks ago I was rooting about in the AWS Marketplace and stumbled across this low-code solution,” he said.

“Is that how you’ve been crushing your story points?” I asked. We don’t like to measure developer performance on story points or productivity - it’s team performance and value flow that matter after all. We use story points so that Pat can properly set expectations at the portfolio level about what we are likely to

deliver when. But Jason's productivity had spiked in the last sprint and he hadn't been going home any later than usual. It hadn't gone unnoticed in the sprint review, but he'd just laughed it off and blamed it on the new coffee machine. He nodded.

"How come this is the first we've heard about it?" I asked. We create plenty of opportunities to share - the daily standups and the reviews and retrospectives for starters, and we're colocated again so it's not like we have to Slack everything. Water cooler conversations happen.

"I guess I was a little embarrassed," he replied. "You know, I've always been a bit of a snob about low-code, no-code. It's for people that can't code, right?" I nodded now. We developers can have a bit of an ego about our coding skills. And quite right too - it's not a skill everyone has and is hard-earned.

"Same, same," I said. "The general view is that they are also quite restrictive though... difficult to customize."

"You notice any problems with the feedback on my user stories from the last sprint?" Jason asked. I shook my head.

"This tool does let you get into the underlying code if you want to. I didn't need to, as it happens, but it's an option. A couple of things you'll really like though. Firstly, knowing how much you like AI, it uses it to test and manage the dependencies."

"That darn warehousing integration," I murmured, looking at his screen where he was showing me the IDE.

"Exactly. And, it has its own CICD capabilities but also has integrations into all the tools in your DevOps toolchain. Including Moogsoft - so we can get performance data and feedback way before we even try to go into production." He took a sip of his coffee.

"Our DevOps toolchain." I corrected.

"Team tool, your masterpiece." We smiled at each other.

“You have been a busy bee.”

“I sure have. Shall we have a go at the breed-based community wish list feature now? I thought we could do some pair programming today. Actually, it’s more like trio programming – Pat’s joining us too.”

“I’m sorry?” I said. Pat’s the product owner as I said, not a developer.

“I know what you’re thinking,” said Jason. “But she said herself it’s all hands on deck. She studied computer science and tinkers away with a bit of Python on the side for a charity project she’s involved with. She’s got enough skill to do some damage with low code. Good damage, that is,” he qualified.

They weren’t the shortest few days, but I did get to know both Jason and Pat a lot better. And we had a hoot. And most importantly, in the sprint review, we were already looking at the results from the release of the new feature. We’d only done a canary release to around 5% of users less than 24 hours earlier, but we could already see via Moogsoft it was holding up well as our pre-production analysis had suggested. There was a definite spike in users and that feature itself was seeing some strong session times and transaction conversion rates. Can’t wait to tell my mother how our experiment worked.

## CHAPTER 8

# James Embarks on a Service Desk Migration to Improve Incident Management with AIOps



**IT'S BEEN A MONTH** since Dinesh and I humbly high-fived leaving the meeting with Charlie and Lucia and they gave us the green light to roll Moogsoft out across the whole of C&Js and I'm feeling a little weary. Change is hard. I've also made it harder on myself by persuading Charlie we should also migrate our service desk solution.

We've been using our existing one for as long as I can remember and some of the problems we have with it aren't really the technology itself; they're more about our specific implementation and our rapid consumption of heavyweight IT service management processes in the past. Charlie justified this major change on the basis that he's sick to the back teeth of hearing people complain about it and it's a hefty lump out of his annual IT budget. He keeps calling it technical debt, which I'm not sure is quite right, but I'll go along with it as a justification. My motivation is more that I've seen how Animapanions is using a solution that ties right into its backlog so the line between a new requirement and a support ticket becomes completely blurred. As Sarah explained to me, it's a cycle. Problems so often need to become backlog items and included in the sprint cycles, it makes sense to have them tightly, almost invisibly integrated.



And we get full traceability through the value stream so we can measure cycle time and other flow metrics.

Also, they showed me their integration into Moogsoft and how she was now creating and updating service desk issues from an open Moogsoft situation and a very cool auto-assign feature. We've found it hard to set up workflows and integrations with our own solution, but with this one, it's pretty much just a webhook.

So it all made a lot of sense until I had to get other people involved. I am officially now a herder of humans. I cannot tell you how many times I have had the same conversation in the past two weeks. And how often I've heard the same statements from people who almost in the previous breath said they would rather die than continue working with the existing system. Take this one with my counterpart supporting the finance system for example. Jeff has been with C&Js nearly as long as I have and has some serious ITSM knowledge and skills but, well, let's just say we haven't always seen eye-to-eye. And I was having almost the same conversation in nearly every department.

"But we've always done it this way," he said. "If it ain't broke, why fix it?"



I took a deep breath, trying not to sound rehearsed even though I'd already said the same thing to umpteen people already that week.

"Well, what if there's a better way. What if we could identify a system so fast that we could fix it before it even became an incident?"

"Ha, that's crazy talk. How can we know something's broken before it's broken?"

"Because we can see when it's starting to break."

"But if we know it's going to break, why haven't we already fixed it?"

"Because it's the first time it's happened. But the system's clever enough to tell us that it's seeing something unusual that we should probably look into. For those things that have broken before, if we can't fix the underlying issue, we can use automation to trigger remediation. But we want to be careful with that as it's just like a sticking plaster really."

Jeff agreed he would spend some time with me looking at a demo so I started setting up a pilot for him and his team. The demo could also show him how to migrate at the same time. I was just doing some final checks when I saw an alert pop up in Moogsoft and also saw Jeff frantically waving to a couple of his engineers, shouting at them across the office that there was a severity one incident.

I picked up my laptop and wandered over to his desk.

"May I assist?" I asked, as he frantically clicked on dashboards on his multiple screens.

"Sure," he said, not even looking up. "Just about to create a ticket."

"I've already done that," I said. Now he looked.

"It's just one click from the alert in Moogsoft," I said, and showed him. He opened his mouth to ask him something but I cut him off.

"Let's not waste any more time on that now though, if it's a severity



one, which it looks like it is.” I nodded at the swathe of red across his monitoring dashboard. “We should focus on solving it.” I pointed at the alert on my screen.

He was distracted though, by one of his support engineers shouting from his desk:

“The APM’s saying it’s the application. It’s lost connectivity to the BACS service running in the Austin data center.”

Then his network engineer chimed in.

“We’ve lost one of the firewalls in the Austin data center.”

“Well, which is it?” asked Jeff as his mobile started ringing. I could see from the caller display it was our CIO, Charlie. He took a breath and picked it up, answering it. Charlie, amiable as he is, is not at his most fun to deal with during a severity one incident. He becomes almost robotic and demands fixes that match the SLA regardless of what’s going on behind the scenes.

“Charlie,” said Jeff. “Yes, we are on it. Not yet, no. We can’t be sure of that. Well, we’d be hopeful of that but since we don’t know what it is yet—” I could hear a raised voice and was certain I caught the name of our CEO. There was a reason Charlie couldn’t take incidents lying down. He had his own boss and his pay was tied to system performance. “Um... actually, he’s already here,” Jeff swung in his seat to look at me. “Ok. We’ll call you back.”

“That was Charlie. He said to ask you to help. And also we have to fix it in the next 22 minutes.”

“Take a look at this,” I showed Jeff the Moogsoft alert and beckoned the engineers over. “Those alerts you are seeing are correlated. I’m certain they are part of the same incident. Can we take a look at the firewall?”

“Sure,” the network engineer said, as Jeff brought up a new screen. “I just manually restarted it.”

As it came back online, we saw the alerts fade on the screen. “I’m guessing the firewall being down blocked application server connectivity in the data center. Any ideas why?”

“I’ll close the ticket,” said Jeff.

“Well, let’s wait a moment until we have an action from this, which we’ll call our retrospective,” I suggested.

“Ok, but we only have 17 minutes left on the SLA,” said Jeff.

“Well, it’s a little behind on its patches,” said the engineer. “I’ve had an open ticket in the backlog for weeks now to update it.”

“Prioritize that and do it next,” said Jeff. “Wow. We really have to concentrate on getting that technical debt paid down. We should really get some patch management sorted too. There’s just so much going on, all the time.”

“I’ll update the ticket,” I said, showing Jeff how to do that, and how closing it in Moogsoft closed it in the service desk too. “It’s

bidirectional, so no manual duplication for you. That should save you a bit of time.”

“It will. And I like how it helped us see why we needed to address that particular issue. It’s by no means the only upgrade we are lagging on. Or the only type of technical debt we’re suffering from. Justifying spending time on it’s always really hard though.”

“I bet!” I said. “You can look for trends though quite easily now on how often it’s causing you a problem. And the time you’re saving by finding the problem faster can let you. Eventually, you’ll find yourself in a place where you have enough time to focus on improvements as well as unplanned work, BAU, and paying down technical debt.”

“That’s the dream,” said Jeff. We completed the rollout of the new service desk with Moogsoft integration in the finance department later that same month.

## CHAPTER 9

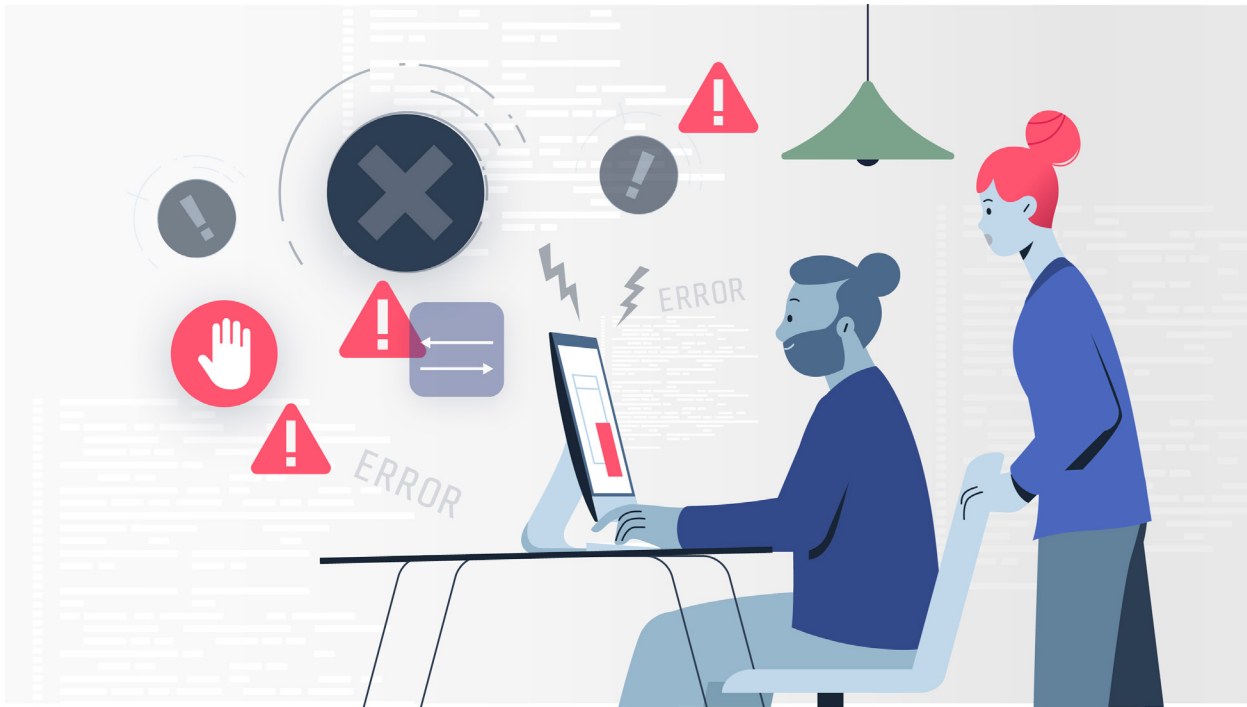
# Dinesh Experiments with Chaos Engineering



**A** **NOTHER DAY**, another drama! This one, though, is very much of my own making. I have been wanting to try my hand at a bit of chaos engineering for some time now but C&Js just hasn't been ready. Sarah's been up for it too, though, at Animapanions. And now that our CIO, Charlie has seen MTTR drop across every single technology team, thanks to the rollout of Moogsoft and the new incident management system (kudos to James), it's pilot day.

Talking of pilots, my old man is a retired civil aviation captain and I've found it really useful to talk about the work he does when trying to convince people that chaos engineering's a valuable practice. I was chatting with James's nemesis, Jeff, in the finance department the other day. The conversation went a bit like this:

"Um. So I really don't understand why I would break the thing I spend my entire life trying to keep running, on purpose." Jeff did look genuinely baffled to be fair. It was clear some serious neural pathway reengineering was on the cards.



“There are a lot of similarities between aviation and information technology. They are both highly complex environments and when something inevitably goes wrong, it can be catastrophic and is rarely the cause of a single factor.” I knew that chatting with Jeff about airplanes was going to catch his interest. He’s into aerobatics in a big way. But, let’s face it, everyone wants to know how to be safe when we fly.

“That’s why we work so hard to maintain our aircraft. And our systems,” Jeff said, wisely, I will acknowledge.

“True, that. And I’m not diminishing the importance of the constant monitoring and updates we do,” I said, not mentioning the recent outage James had told me about where a severity one was caused by an unpatched firewall. “But this is next level. Belt AND braces if you like. When I was growing up I remember asking my dad why he spent so much time in the simulator. Sometimes it seemed like he was in there more than he was in the air.”

“I do like Blue Angels,” said Jeff, a dreamy look in his eye as he, no doubt, recalled his latest gaming bout. We were both big gamers.

“I pre-ordered the new Flight Simulator last week,” I said, getting slightly off track.

“No way! Me too. London Heathrow here we come!” At least we seemed to be bonding.

“So, my father said to me that every time he was in the simulator, he practiced what to do when something went wrong, so if it went wrong in the air, he’d know how to fix it.”

“Oh,” said Jeff, fiddling about on his phone. I guessed he was googling. ‘Chaos engineering is the discipline of experimenting on a system in order to build confidence in the system’s capability to withstand turbulent conditions in production,’ he quoted. “I see what you’re getting at.” Say what you like about Jeff, he’s quick. “When did you last try this?”

To be fair, I can understand why he didn’t want to be my guinea pig. He does have one of the most notoriously unstable systems in the company. Sarah, on the other hand, has her MTTR at its lowest ever, and the lowest in the company thanks to Moogsoft and her DevOps toolchain. She’s also practicing



a new swarming technique and looking for ways to tackle the technical debt in her own system as she still does get a surprising amount of incidents.

Chaos engineering isn't just a fire drill - it's also a way of finding underlying problems and technical debt and fixing them before they cause an incident.

Sarah had her own pertinent analogy for chaos engineering that I enjoyed too. She said she thinks of it like a vaccine, where you inject yourself with a small amount of a potentially harmful foreign body in order to build resistance and prevent illness. She said it's a tool we use to build immunity in our technical systems by injecting harm (like latency, CPU failure, or network black holes) in order to find and mitigate potential weaknesses.

She arrived at her desk, surprised, but I think pleased, to find me waiting in her chair.

"Excited, much?" she asked, handing me the tea she'd picked up for me. I like Sarah.

"Absolutely! This is going to be great — a game-changer."

"I know! The swarming team is assembling. The chaos tool is running. Moogsoft is ready and tuned. We have our hypothesis and experiment defined. The 'what could go wrong?' exercise is complete." Sarah waved at the whiteboard we'd all been working on yesterday. "I think we're ready to go."

"And we're experimenting with a 'known known' today?"

"We'll start there, yes. We need to start with something we are aware of and understand. Then we'll move onto the things we know about but don't really understand and work our way up to things we don't know about and don't understand."

"That sounds sensible." We grinned at each other.

"Ok," Sarah said, calling her team together in the manner of a daily stand-up. "Rachel, please can you remind us of the experiment we decided to run today?"



Rachel is Sarah's lead support engineer and used to work in the network team at C&Js. You can see how much she's blossomed at Animapanions. She's been given space to grow and is rarely constrained by the bureaucracy that really used to get her down.

"Today, we're going to test the hypothesis that the network is not reliable. We know this to be true, not just because we've observed it many times ourselves," Rachel paused as her teammates chuckled, "but because one of the fallacies of distributed computing is that the network is reliable."

"Thanks, Rachel," said Sarah. "This is an example of one of those dependencies that are out of your control, and sadly always will be. We can't follow our DevOps mantra of 'don't manage dependencies, break them' here so it's really important we understand how the system reacts when it is unavailable and do our best to shield our customers."

"Totally," said Rachel. "So we're going to do a network black hole chaos experiment that will make the designated addresses unreachable from Animapanions. Once we've applied the black hole, we will check if we can start up normally and serve customer traffic without the dependency. We'll also be keeping a close eye on Moogsoft to see what else it can tell us about what's happening."

"Exciting stuff!" said Sarah. "What results are we expecting?"

"We think the traffic to dependency goes to 0 (or gets slow), startup completes without errors, application-level metrics in steady-state are unaffected, traffic to fallback systems shows up and is successful, dependency alerts and pages may fire. We are scoping this to a single instance."

That's pretty much what happened. It looked like a switch was gone and the SDN rerouted successfully. We could see it all in Moogsoft.

Rachel had baked us a cake for the retrospective.

“That was all good,” said Sarah as she sliced into it. “Remember though, success and failure should both be celebrated. Failure is a learning opportunity. As long as we protect ourselves from catastrophic failure that impacts our customers and harms our business and ourselves, whatever the result, the experiment is useful.”

“Exactly,” I said. “That’s why we limited the blast radius of that experiment and were ready to switch back to the blue environment if we needed.”

“When can we do it again?” asked Rachel.

“What do you want to do next?” asked Sarah. “We could do the network again? Test the latency, packet loss, or DNS? See what happens with DDOS? Or we could do something with resources like CPU, memory, IO, or disk? Or we could go really chaotic and look at state. Shut down, time changes, or killing processes, anyone? Now, if only there were a way to do this continuously,” said Sarah, thoughtfully.

## CHAPTER 10

# Sarah Resigns from Animapanions and Heads Off to a Competitor



**HAD MY LAWYER** check over my employment contract first, and thankfully it was missing a non-compete clause. It's not like I was super unhappy at C&Js but I just work better in smaller companies. I've always worked in start-up environments and I love the scrappiness of just getting stuff done. C&Js had made the right noises about not wanting to crush our entrepreneurial spirit at Animapanions and using their scale to unleash levels of creativity, but I think it's impossible not to get sucked into the Borg.

I saw it starting when they rolled our warehousing and delivery services over to the central C&Js systems. Yes, that gave us a fast track to international deliveries, but it also created a dependency and a handoff that has created more problems than it solved in terms of delays in the flow of our value stream and unplanned work (i.e. incidents).

Then there's this sense that I'm spending more time educating the rest of the business now than investing in Animapanions. I'm flattered, don't get me wrong, that I'm seen as some kind of guiding light to the new ways of working. But I'm also frustrated, because I know I could be doing more if I had more time to invest in it.

So, once more in my career, I wrote that bland resignation email and pressed send. As expected, I was called in for an exit interview with HR. “Sarah, we are really sad to see you go. As you know, one of our key performance indicators is staff retention. We’d love to learn how to avoid letting talent like you slip out the door in the future.”

It costs a lot to attract, train and retain staff. The price of replacing me is at least double my annual salary. I know this, and there is a sense of letting my colleagues down, but sometimes, we have to put ourselves first in life. I know this is the right decision for me - emotionally, far more than financially.

“Would you be willing to share why you are leaving your current position?” Tag asked.

“Of course!” I said. “It’s just I’ve been offered something I can’t refuse. It’s a great opportunity to be involved at the start of something new.”

I didn’t say it was better paid as it wasn’t. But it did come with share options which are pretty light on the ground at C&Js, it being such a large organization. Tag was nodding.



“Do you think you were fully equipped to do your job well?” I was happy he’d moved on from the last question.

“Sure,” I said. He frowned at me. I thought I’d throw him and all the people I was leaving behind a bone. “Eventually, I mean. It could be painful to access the technology we wanted to experiment with to improve our product. Much more so than before the acquisition. I understand there needs to be governance and policies to protect us, the business, our customers, but in a high-velocity environment, often we need to just try a piece of software now. Not wait weeks for approvals from procurement or security or whoever.”

He made sure to write all of that down. “How was your relationship with your manager?” he asked. This was an easy one.

“Perfect,” I said. Tracy and I went way back. I didn’t even think of her as a boss. We’d known each other a long time before C&Js came into our lives and I was certain we’d know each other a long time after too. In fact, we’d already had chats about her following me to the start-up. She had her own frustrations, mainly around the layers and layers of hierarchy she now had to navigate.

“What is it that’s so great about working with Tracy?” Tag asked. I knew that Tracy hadn’t been through most of the management training C&Js expected from their leaders. She hadn’t been prioritized as she was too low down in the pecking order. Also, she didn’t have a lot of time for their approaches. She didn’t consider them aligned with our new ways of working values.

“Tracy is a transformational leader,” I replied. “She’s much less a manager and much more a coach and mentor. She helps us discover ways in which to make improvements and gets things out of the way. I don’t think we’d have managed to get Moogsoft in or architect and build the DevOps toolchain without having her fighting for our corner.” I didn’t let on that she had actually found ways of doing things under the radar, cutting governance corners just to let us get on.

“What was the biggest factor that led you to accept this new job?”

Tag went on.

“Sometimes it’s just time for a change,” I said. “You know, I’ve only been at C&Js for about a year and a half, but I was at Animapanions from the start. That’s nearly six years.”

I didn’t add that I wanted new skills, that I’d learned so much over those years and so much had changed in the technology landscape and I wanted the chance to take what I’d learned about observability and DevOps toolchain architectures and build something from scratch with no barriers, no restrictions, no bureaucracy getting in the way.

“What did you like most and least about your job?” he asked. I wanted to answer, “the people and the people”, but I didn’t. The people I didn’t get on with so well weren’t necessarily bad people. They were just conditioned to certain ways of working, found change frightening, and were trying to work within the rules they’d been given. They were doing their best.

Instead, I said: “The opportunity to learn how large organizations operate, and the opportunity to learn how large organizations operate.”

Tag smiled and asked me to elaborate.

“Well,” I said, thinking, annoyed my flippancy had led me down this path. “It’s really amazing to see so many humans working together, systems that have been built at such scale. How over hundreds of years we’ve reached a point where all these products can be manufactured all over the world and reach the kitchens and bathrooms of all these people anywhere.”

He looked at me, pressing me to continue.

“But the flip side of this is sometimes it feels like the system we’ve built is kind of toxic. I mean, not on the scale of The Big Short or Spotlight or Making a Murderer, but we know we aren’t doing everything right and it’s such a huge effort to make change happen

- at the C&Js level I mean.” I think I may have lost him there and he moves on to his next and final question. I breathe a sigh of relief. Freedom is just around the corner.

“What skills and qualifications do you think we need to look for in your replacement?”

“It’s more about attitude and character,” I suggested. “If someone’s open to learning, they can be taught anything. They can teach themselves. They’ll need some basic understanding of how to code, but you don’t necessarily need an expert programmer. They don’t need data science skills as we have Moogsoft. It would be useful if they’d used a DevOps toolchain before, but the team has the blueprints and is using it all day every day and rolling it out to the rest of the organization.”

I am happy I am leaving them the blueprints, but even happier I’m taking them out the door with me too. Not literally of course, but figuratively. They’re in my head. I’d architected, built, and used them of course. They are part of



my DNA now. I'd learned from the mistakes I'd made, fixed broken pieces, built essential integrations, tested and tested them under the weight of daily business. We are getting a head start, thanks to this experience and the skills and knowledge I'd gained at Animapanions. Our vision of a luxury feline lifestyle brand, Catucci, is poised for launch and we have everything we need to scale it from the get-go and just focus on those features that will delight the ever-growing customer audience of humans devoted to their cats.

We've already got our cloud instances ready and I've already set up the end-to-end DevOps toolchain using the latest and greatest tools, with an API first approach so we are maximizing the opportunity for traceability and optimizing cycle time. Moogsoft will be able to gather data from planning and backlog, all the way through CICD, through the service desk, and into production, so we can fix an incident not only fast but also hopefully see anything unusual coming and keep technical debt out of our way. I've never been so excited about starting a new role.

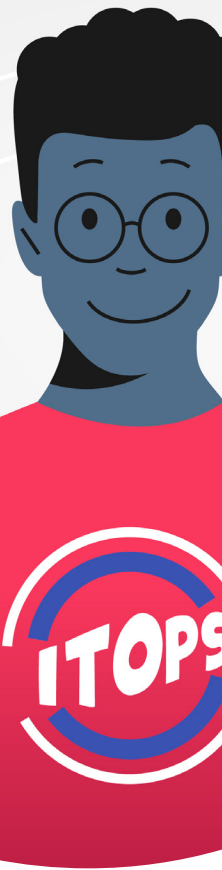


## CHAPTER 11

# James Speaks with the Industry Analysts

Gartner.  
**EMA**

FORRESTER®



**C**HARLIE, OUR CIO, is leaving. That bombshell hit me yesterday afternoon, shortly after the bombshell that Sarah's leaving. Sarah, I'm really gutted about, Charlie, maybe not so much. Partly what's put my nose out of joint is that his replacement has already been identified and is already here, in an "interim" capacity. Apparently, Charlie had a big old row with the CEO, and one of the CEO's old college pals, a management consultant called Derek, has been helicoptered in to usher Charlie out the door lightning-fast.

I'd like to have a crack at the CIO role myself but I'm guessing there are reasons I've been stuck in middle management all these years, despite my best efforts to drag myself up the ladder, and I'm not sure those reasons have changed. Criticisms that have been leveled at me in the past are that I don't think strategically, I'm stuck in the weeds and the details. Oh, and also that I'm a bit hot-headed.

Not a lot I can do about that last one, apart from maybe playing a neuro-diversity card – it takes all sorts, right? Also, I can't remember the last time I had an all-out shouting match with someone. The meditation must be

working, that or age is bringing me wisdom. Anyway, if the CEO can be heard in the car park letting off steam at one of his team, why should there be another rule for the rest of the staff?

The root cause of all the yelling? Money, of course. Isn't it always? It's a bit of speculation but the snippets of what was heard during the "incident" between CEO and CIO have been pieced together in a cross-departmental post-mortem type gossip fuelled activity (all very agile and collaborative actually) and it boils down to Charlie being told to do more with less. To accelerate the digital transformation with no additional investment. Actually, expect budgets to be cut.

On reflection, I warm to Charlie a bit. I always thought he was a bit of a yes-man that would make promises on our behalf that we couldn't possibly keep and yet it would all be our fault when it didn't work out. It seems, with his back against the wall, he fought our corner and it cost him his job. No doubt he'll find a better one though, so back to me.

That other criticism - that I'm not a strategic thinker - it's time to tackle it. The CEO has tasked Derek with overseeing a review of all IT investments. It's time to slim down. Take a machete to that supplier list. I've managed to get



into a position to kick the process off, looking at our AIOps space. Charlie's never had a problem funneling vast amounts of cash into the industry analysts' pockets. Perhaps I should suggest Derek to review that spend, but in the meantime, I'm going to have a chat with them and show the CEO what a strategic thinker I can be. And hopefully, defend and protect the investment Dinesh and I have made on C&Js behalf in Moogsoft.

First up, Gartner. I asked them to walk me through their AIOps market guide. They alerted me to this statement:

“There is no future of IT operations that does not include AIOps. This is due to the rapid growth in data volumes and pace of change (exemplified by the rate of application delivery and event-driven business models) that cannot wait on humans to derive insights.”

I am known to be cynical, the posture of the weary, but even my past dismissals of analysts as pundits pedaling futures on behalf of vendors, can't stand up to this statement. It's so blindingly obvious. With the huge amount of data created by machines, it's impossible for humans alone to consume it, understand it, and make decisions based on what we see in it, without help from some more machines. Then they said this:

“AIOps platform adoption is growing rapidly across enterprises. I&O leaders are planning for a post COVID-19 environment dominated by practical outcomes rather than aspirational goals.”

Biting my tongue at what seemed like marketing speak here, I took three deep breaths and donned my strategic thinking hat.

“Can you clarify what you mean by practical outcomes over aspirational goals?”

“Sure,” said the analyst. “In the past, we've heard organizations saying that they want to reduce MTTR but they haven't really expressed why. Now we're hearing people understand that noise reduction directly translates to less time on unplanned work and more time on paying down technical debt and making improvements and enhancements.”

Cool. I thought. That’s exactly what Dinesh and Sarah had been teaching me, and they’d been helping the metrics that we used in the business case for the enterprise rollout of Moogsoft. I already had half of the report then for this one. A perfect example of doing more with the same. Now I needed some ammunition for doing more with less. They’d got me a bit lost in the report around their “domain-centric” and “domain-agnostic” classifications of vendors, so I just asked them which one we should be buying. Not pointing out that we already had instances of nearly every vendor’s product mentioned in their market guide somewhere in C&Js.

“If you have a requirement for increased flexibility for processing highly diverse datasets then you probably want to look at domain-agnostic functionality. How much variety do you see in your data?”

Loads and loads we agreed. So another massive tick for Moogsoft. And also, my opportunity to do more with less. It seems we don’t need all the tools we have. I made a note in my task list to drop into procurement later in the week and ask them to run a report on the annual licensing fees for all of the monitoring tools we use.



Next on my list of analysts was Research in Action. Their research had really got me scratching my head when they tried to rename the entire market category:

“Bottom line is that the future lies in leveraging AI’s power to predict across application development, IT operations, and service management which is why Research In Action has decided to rename the AIOps research into AI Predictive Analytics.”

I’ve had conversations with Dinesh umpteen times about prediction and we always agreed that if you could predict an incident you’d already failed – you should have already fixed it. But Gartner had said something similar too:

“As organizations continue to undertake digital transformation, they no longer have the luxury of responding to issues after they occur. Instead, they must become proactive and address potential issues before they impact user experience.”

Forrester helped me answer some of my questions in their recommendation:

“Provide deep end-to-end digital experience monitoring (DeM). Functionality such as real user monitoring (RuM) and synthetic transactions have been around for many years. Two things have driven these capabilities from nice-to-have to mandatory. The first is the massive increase in the remote workforce due to pandemic-related protocols; the second is the dramatic rise in demand for a wide variety of online digital services. Enterprises must now be able to tie the experience of a digital user to business outcomes. Unless they can relate user performance and sentiment data to business KPIs, operations and product teams are flying blind.”

It seemed what we could do with this technology was significantly more than just incident management. This got my newly nurtured strategic thinker brain excited. Another example of doing more with less (or at least the same)?

My final research analyst, EMA, confirmed it:

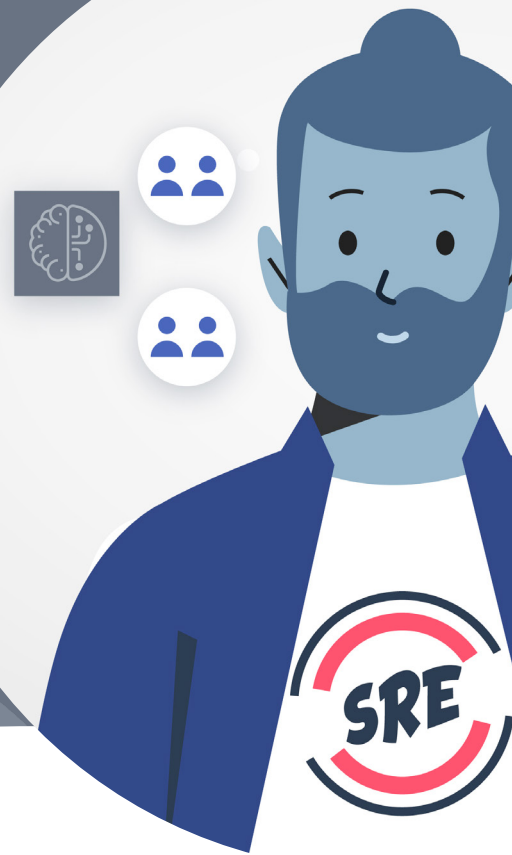
“We see three AIOps use cases. The first is incident, performance, and availability management. This focused on optimizing the

resiliency of critical application and business services - including microservices, VoIP, and rich media - in cloud (public/private) as well as non-cloud environments with a strong focus on triage, diagnostics, roles supported, self-learning capabilities, and associated automation. Next, we have change impact and capacity optimization. These are admittedly two use cases combined into one, but share requirements in terms of understanding interdependencies across the application/service infrastructure as volumes increase, changes are made, configuration issues arise, and automated actions are required. And thirdly, business impact and IT-to-business alignment. This includes user experience, customer experience, and customer management, business process impacts, and other relevant data, with an eye to supporting business initiatives, such as digital transformation through superior IT-to-business alignment.”

This then is my killer move. Watch out, Derek. I’m going to show the CEO how the investments he’s already made on my recommendation have allowed teams to innovate faster, how he can throw away a bunch of our older monitoring tools and save on license fees, and how he can finally connect IT and the business. **IT doesn’t align with the business, or integrate with the business - it IS the business** and this is going to prove why and how. He’ll finally have the technology-driven business that will see us thrive through the age of software and become the disruptor, not the disrupted. He won’t want to be doing more with less - he’ll want to do more with more when he understands we need to be customer-driven not shareholder-driven to succeed. And that is some strategic thinking.

## CHAPTER 12

# Dinesh Starts an AI Community of Practice



**T**HERE'S AN OLD ADAGE IN DEVOPS, “Give the developers a pager and put them on call.” As an SRE I find this interesting. One thing, we don't have pagers anymore, we all have smartphones so this seems like something prehistoric. Another thing, are we trying to punish our developers? Slap them into learning something? It doesn't sit well with me, I've been both a developer and an IT ops guy - it's what, I'm told, makes me a perfect SRE. In our new world, we build it, we own it. We're aiming for autonomous, multifunctional teams - not silos handing off to one another.

Anyway, rant over. For now. I mention all of this because Derek, the new, interim, CIO said it to James and I the other day. I actually thought James might explode. He was showing Derek and the CEO the research to support our investments in AIOps (or predictive analytics?) and there was a lot of nodding up until that point. Let's just put it this way, the pair of us are very hopeful that Derek is very much interim and someone else arrives who's somewhat more forward-looking. For example, like James, who has become remarkably protective of his developer friends and I applaud that.

In our “debrief” (read: three beers each and a lot off our chests) I gave myself an action item to set up a community of practice. We already have a few for agile, DevOps and cloud and they’re going quite well. I’ve dropped into them and what impressed me was that a bunch of people are interested enough to voluntarily turn up and make things happen. They’re a little bit underground at the moment, not CXO-approved if you like, but people in the middle-management are supportive. I like anything a bit subversive and this smacks of the grassroots movement we need. If we’re going to be successful in this disrupted digital economy, we must have autonomy. We can’t exist in a command-and-control culture and be empowered to participate in change. We’re taking the reins.

My community of practice, of course, is going to be AI. What I’m hoping is to tap into a whole host of humans I don’t know but share my interest and also bolster James’s run for president. I mean CIO. I think he’d be amazing. That guy knows how to get stuff done. He’s not bothered about being everyone’s mate. He takes no prisoners. And he really, genuinely cares about doing things right and can see the future. He has some serious vision and he inspires me. I like working with him and if he gets that role, it could mean great things for me too as we are genuine buddies and I know he likes the way I work.





So when he said the AI community of practice was a good idea, I went all-in on it on the strength of his blessing. Here's how our inaugural get-together went, two weeks after I posted the plan and date across a bunch of company Slack channels and told everyone in hearing range what I was up to.

It was an experiment, so we had a hypothesis. A simple one: we build it, they will come. We said twenty people would turn up to the meeting. We made it a Friday afternoon in the office and I forked out for donuts. We were going for the fun and friendly vibe. Forty-eight people turned up, so the experiment was considered a success.

We should probably have had a hypothesis around outcomes too and I was kicking myself for the omission, as it's become a bit of a mantra to James and I: outcomes over outputs. The number of people is an output. I had a bunch of topics I thought we could talk about, but I also wanted to let the community decide what they wanted - to let it be self-organizing.

It's not that I regret that decision, but we did go in a direction I wasn't anticipating and wasn't prepared for. Enter, Duena. First impression? I was a little scared, I'm not embarrassed to say. Also, how have I not met Duena before? She is a force to be reckoned with. And obsessed with AI. I feel like I've met my match and, let's put it this way, not all of my thoughts are strictly professional.

Duena came along with an axe to grind, and the donuts weren't calming her down. What was bugging her was PagerDuty. Specifically, how she felt it was still too noisy, despite the salesperson there telling her their event intelligence was sorting out the constant stream of alerts she was receiving.

This was why I hadn't met Duena before. She works with one of our very recently acquired products and this was her first time in our new offices. She'd come in especially for this. I don't mind saying I was flattered. But it's not like it was about me, it was about the topic.

I'm getting off track. We don't use PagerDuty anywhere else in C&Js so this was new. Did I tell you she's also an SRE? It just gets better and better. Even better when a quick Google search showed me I had a prime opportunity to gallop in like a white knight on a stallion. I had a feeling she probably wouldn't



like that analogy so I kept it to myself. When she looked like she'd set out the fullness of her complaint, I stepped in.

“Duena,” I said, such a nice name, I’d never said it before. “We’ve been experimenting with an AIOps tool called Moogsoft for quite some time. Way beyond experimentation really; we’ve got quite the portfolio of outcomes and evidence for the improvements it’s made.” James nodded, sat next to me. “When Moogsoft surfaces an incident, it can send it to PagerDuty in real-time. Based on the insights derived from the underlying data through the algorithms applied by Moogsoft, PagerDuty knows which teams and people need to take action and those who need to be kept up to date.

“Users have context and can acknowledge and escalate and can add comments and notes directly from PagerDuty. The Moogsoft AIOps Situation Room allows all users to share a consistent view, while both platforms stay in sync throughout the lifecycle of the incident. Once the incident is resolved, PagerDuty streamlines post-mortems to speed up future response, by leveraging Moogsoft’s historical knowledge of prior, similar incidents. It

sounds like we could make this experience a lot better for you. Really get that noise out of the way.”

“Sounds great,” she said, putting her glasses on and opening her laptop.

“No time like the present!” I said. I liked this. This is how I work too. I got my laptop up and running too. “So the Moogsoft and PagerDuty integration enables bidirectional communication between Moogsoft and PagerDuty. The integration allows you to send events to PagerDuty from Moogsoft Enterprise alerts and Situations. Each event relates to a service that creates a PagerDuty incident. Notes you add to a PagerDuty incident appear in Moogsoft Enterprise as collaboration posts in related Situations. Likewise, posts you add to a Situation appear in the PagerDuty incident notes.”

“That all sounds straightforward,” Duena said. “I’m assuming you’ll install this as the Moogsoft admin?”

“Yes, I’ll drive it but we both have things to do. Do you have the user role and an API User Token for it? We need that to query services for integrations, create integrations and extensions via the API.”

“Check,” said Duena.

“You also need to have the responder team role to create, acknowledge, and resolve incidents.”

“Also check.”

“I’ll add you as a user to Moogsoft and give you a role that can access alerts and make sure the situation tools have the “moolet\_informs” permission.”

“Received and logging in,” she confirmed. This lady is quick. I spun my laptop around and moved closer so she could see the screen.

“These integrations are so fast. First, we go to the Integrations tab. Then find PagerDuty in the notification and collaboration section. It

needs a unique integration name - we'll just use the default. Now you add the connection details for your PagerDuty system and we'll configure the service mappings." As she leaned in to use my keyboard, our shoulders touched for a moment.

"I'm just going to enable it to automatically create integrations and extensions and see what happens. We can tweak it later, right?" She looked at me as she asked. My heart skipped a beat.

"Of course. The integration creates extensions and integrations with the name 'Moogsoft-Integration' and updates the URL of any existing extensions of the same name to this value. You should find this sorts out your noise problem and you'll probably end up finding out a lot more about your systems too. I'd love to hear how you get on. Maybe at our next community of practice meeting?"

It turned out that wasn't going to soon enough. We're going to dinner Saturday night. She wants to know more about my AI Ph.D. I want to know more about her.

# About the Author



## HELEN BEAL

Helen Beal is a DevOps and Ways of Working coach, Chief Ambassador at **DevOps Institute**, and ambassador for the **Continuous Delivery Foundation**. She is the Chair of the **Value Stream Management Consortium** and provides strategic advisory services to DevOps industry leaders such as **Plutora** and **Moogsoft**. She is also an analyst at **Accelerated Strategies Group**.

She hosts the **Day-to-Day DevOps webinar series** for BrightTalk and the Value Stream Evolution series on **TechStrong TV**. She speaks regularly on DevOps and value stream-related topics, is a **DevOps editor for InfoQ**, and also writes for several other online platforms.

She regularly appears in TechBeacon's DevOps Top100 lists and was recognized as the Top DevOps Evangelist 2020 in the DevOps Dozen awards and was a finalist for the Computing DevOps Excellence Awards DevOps Professional of the Year 2021.

## Resources

Check out these webinars at [moogsoft.com](https://moogsoft.com):

[Intelligent Observability: Blamefree Retrospectives >](#)

[Intelligent Observability: What The Analysts Say >](#)

[Intelligent Observability: Automation in DevOps Toolchains >](#)

[Intelligent Observability: Making Toil a Thing of the Past >](#)

[Intelligent Observability: Connecting All That Data >](#)

[Intelligent Observability: Under the Covers of AIOps >](#)

[Intelligent Observability for SREs >](#)

[Intelligent Observability for ITOps >](#)

[Intelligent Observability for DevOps >](#)

## Contributors include:



### **RICHARD WHITEHEAD**

As Moogsoft's Chief Evangelist, Richard brings a keen sense of what is required to build transformational solutions. As a former CTO and Technology VP, Richard brought new technologies to market and was responsible for strategy, partnerships and product research. He has served on the advisory boards of Splunk, RedSeal, and Meriton Networks. Richard holds three patents and is considered dangerous with JavaScript.



### **THOM DURAN**

Thom Duran is a Director of SRE at Moogsoft, where he leads a team focused on building the Moogsoft Observability Cloud platform. Thom got his start in the trenches of a traditional NOC and then moved into more traditional SRE roles at GoDaddy. Thom's goals are always about driving efficiency and making sure people excel and love what they do.



### **DAVE CASPER**

Dave is the former CTO of Moogsoft and has 20+ years experience in the CTO/Architecture function at several of the world's largest financial institutions including UBS, Merrill Lynch Bank of America and Deutsche Bank. Dave also co-chaired the Infrastructure Working Group at the Open Data Center Alliance, exchanging ideas on IT and its assurance with some truly great leaders and practitioners.