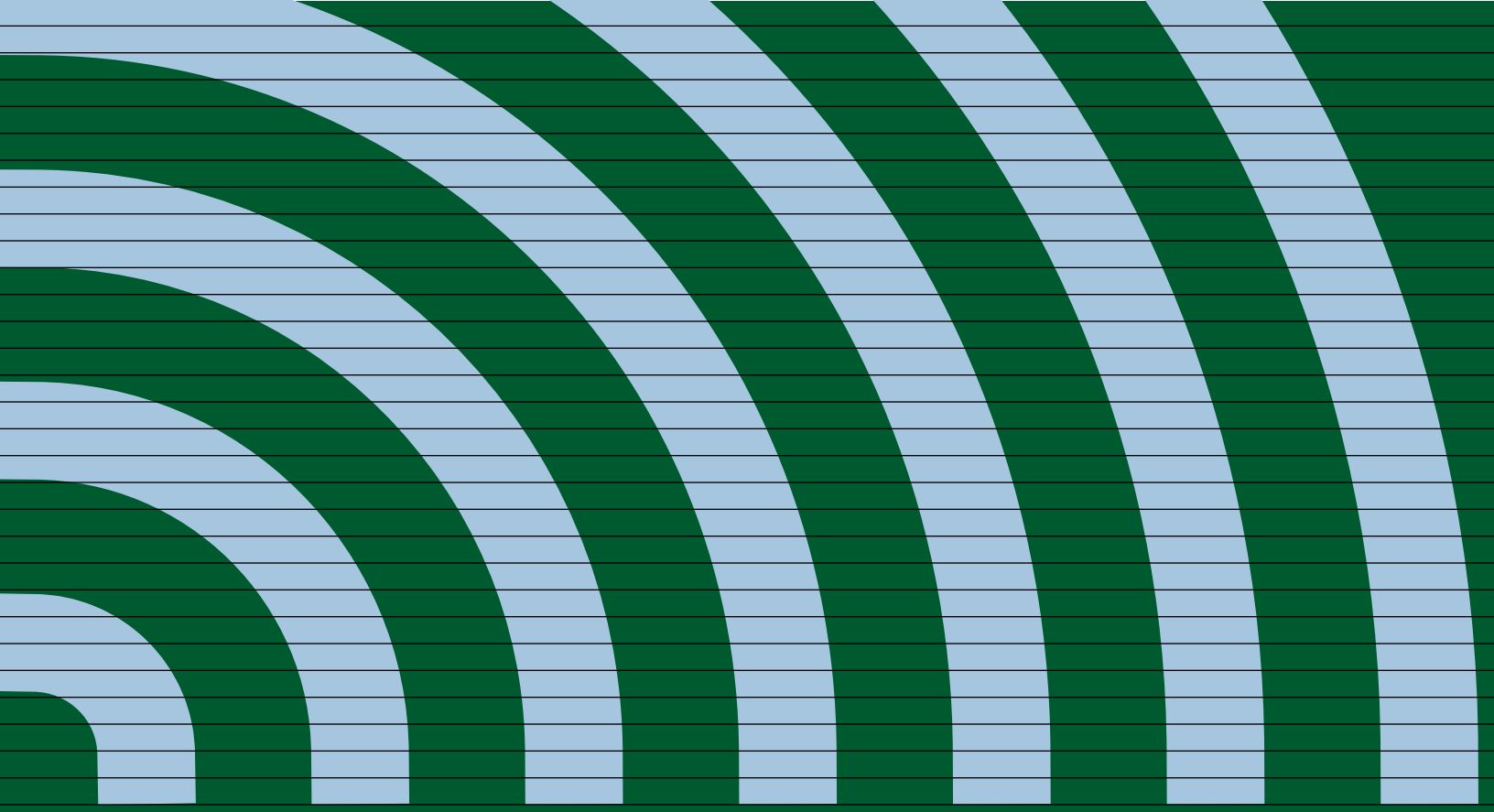

PagerDuty

The SRE transformation

Setting up your organization for success

Table of contents

Introduction 3
Understanding the SRE mission and tactics 4
Exploring the SRE skills and models 5
Culture is key 6
Common SRE best practices 6
How does PagerDuty empower the SRE 7
Looking down the road 8

PagerDuty

Introduction

"Any business that delivers digital products for their customers has a stake in ensuring reliability, stability and incremental improvements."

– Dheeraj Nayal, DevOps Institute

It may be hard to believe, but there was a time not too long ago when the Internet was small.

No smart phones. Little to no social media, at least how we experience it today. And certainly no "big data" or global pandemic that supercharged an already accelerating technology explosion.

This was the world in 2004 when Google, itself a much smaller company than it was to become, foresaw a time when engineers would be drowning in data, forcing them to act with greater agility and employ automation to solve routine problems. Google's nearly two-decades old model has since become the industry standard for service reliability.

We know the practice today as Site Reliability Engineering—an ever-growing movement to accelerate digital transformation and reengineer systems to be more proactive, predictive, automated, and customer focused.

"SREs (the engineers who make it all happen) and DevOps are two sides of the same coin," said Mandi Walls, a DevOps advocate with PagerDuty. "DevOps engineers focus on development and building solutions to cater to business requirements, writing and deploying code—the 'what' needs to be done. SREs deal more with operational problems and have a broader end user perspective, focusing on the 'how' something can be done. They complement each other."

"By 2027, 75% of enterprises will use site reliabilty engineering practices organization-wide to optimize product design, cost and operations to meet customer expectations, up from 10% in 2022." – Gartner, How Software Engineering Teams Should Work With Site Reliability Engineers, Daniel Betts & George Spafford, 23 January 2023

Reliability is essential

According to a recent Gartner report, "customers expect more from the reliability of digital products than ever before. They expect applications to be reliable, fast and available on demand."

"Unfortunately, many products fail to meet customer expectations because software engineering teams are focused on releasing new features quickly and do not prioritize product reliability," the report stated.[1]

The Gartner report further states, "Software engineering leaders must treat reliability as a differentiating feature in delivering business value. Their teams must work closely with site reliability engineers to help the product owner define SLOs, share accountability for meeting SLOs and adopt SRE practices and tools."

This guide will help you better understand the role of SREs and their evolution; how SREs can best fit into your organization; and what the future might look like as you continue your digital transformation and set your teams up for success.

---

[1]Gartner®, How Software Engineering Teams Should Work With Site Reliability Engineers, Daniel Betts & George Spafford, 23 January 2023

GARTNER is a registered trademark and service mark of Gartner, Inc. and/or its affiliates in the U.S. and internationally and is used herein with permission. All rights reserved

Understanding the SRE mission and tactics

The Google SRE model was meant, at least in part, to codify practices that would make it easier to achieve the promise of DevOps and bring the DevOps philosophy to life. In fact, Google's core principles of DevOps and SRE are nearly identical:

- **Reduce silos** – SREs help by sharing ownership across developers and production teams
- **Eliminate blame** – incidents used as learning opportunities
- **Create gradual change** – smaller changes easier and safer to dissect and iterate
- **Eliminate toil** – leverage tooling and automation to do repetitive tasks without human intervention
- **Measure everything** – SREs focus on measuring toil and reliability to make sure that both customers and software teams are happy

The SRE function today

SREs are responsible for availability, latency, performance, efficiency, change management, monitoring, emergency response, and capacity planning. They are another member of the development team but with a different set of skills, particularly around automation and deployment.

SREs typically split their time between developing systems and operations/on-call duties, such as:

- Using automation to minimize human interaction
- Proactively monitoring and reviewing application performance
- Handling on-call and incident management/support
- Ensuring software has good logging and diagnostics
- Creating and maintaining incident or operational playbooks
- Conducting post-mortems and analyze incidents for continuous improvement

"SREs help assure reliability by applying an analytical mind-set and quantitative analysis to reliability issues and incident response," said Heath Newburn, a PagerDuty Distinguished Field Engineer. "It's about 'proactive' engineering and anticipation vs. 'reactive' development—modernizing and automating operations while maintaining a greater focus on the end customer. This to me is maybe the biggest difference in how I look at SRE vs. traditional DevOps, in that SRE is really focused on business continuity and delivering services to the client."

Newburn added that the pandemic and how work is done today, particularly in terms of remote or hybrid situations, has increased demand for SRE functions and skills.

Where there once was the Network Operations Center (NOC) way of the world where there were people in seats and eyes on glass, a hybrid, increasingly virtual mode of working breaks that. Teams have found that they need a different way to manage reliability that's more flexible and accounts for these different operating models, all while faced with even greater pressure on performance and uptime.

Don't knock the NOC

SRE and NOC engineers are not mutually exclusive. The main difference is instead of a "room" the NOC is more about a group of engineers who are building automation, setting policies and practices, and stewarding quality across all teams. The goal is not only to support applications when something breaks, but to make sure customers' expectations are met, and that systems support current and future development efforts without increasing the human workload.

"If you had to start from scratch, you wouldn't start a traditional NOC," Newburn said. "It tends to be very expensive, and teaming SREs with other NOC members is actually a great practice."

Added Eiad Abunimeh, Senior Director of Strategic Solutions at PagerDuty: "We're hearing from NOC directors that automation is a top goal. They need to automate as much as possible and not keep responding to alerts that they didn't need to handle in the first place."

Exploring the SRE skills and models

Whether you are looking to reorganize existing teams or seeking new talent to fill specific roles, how you build out the team is critical. This includes not only the people but also the organizational model that will deliver the maximum value and benefits for your organization.

From a talent perspective, you'll want to be sure that your SREs have the following skillset:

- A computer science or applications development background or interests
- Ability to debug, fix, and optimize code and troubleshooting skills that span applications, networking (TCP/IP), and systems
- Skills in automation, deployment, configuration management, monitoring, analytics and metrics
- An interest in thinking about large scale problems that have a lot of moving parts, as well as in diagnosing or fixing a problem
- Being comfortable with the idea of being "on-call" for their services, whether primary or as escalation
- Ability to stay calm under pressure

In terms of how customers leverage SRE roles, there are several SRE models to consider. The following concepts provide a good starting point:

- Stream-aligned team: these are "you build it, you run it" teams. There are no hand-offs to other teams for any purpose
- Enabling team: helps a stream-aligned team overcome obstacles or detects any missing capabilities
- Complicated Subsystem team: provides support when significant mathematics or technical expertise is needed
- Platform team: a group of other team types that provide an internal product/process to accelerated delivery by stream-aligned teams
- Operation centers (NOC): are transforming L1 and L2 into an SRE to help with process optimization and automation

Enabling or Platform teams are the most common ways to build out an SRE function. Sometimes a Complicated Subsystem team is also necessary – for example, if you are new to the cloud, you might assemble a cloud team to manage that transformation while you get up and running.

Culture is key

Adopting a service ownership model for your organization is about more than just having your engineers take responsibility for their own code – it also requires a cultural shift.

You should start from a place of compassion, shared responsibility, and trust. Individuals need to be empowered to make decisions without fear of being blamed for an incident or issue related to their code.

For service ownership to be truly embraced, this "blameless" culture must extend throughout the entire organization. This will naturally take time and will require buy-in from c-suite on down. Try to start with small and achievable steps, and set the example that incidents are not causes for blame.

It may help to practice this approach in low-risk settings to help build confidence. For example, simulating incidents is a great way for SREs to become better acquainted with managing incidents when they occur for real.

Agile methodologies also can be helpful when implementing a culture of service ownership. Being agile can help teams stay the course toward long-term goals, even when the unexpected occurs. Not only can agile workflows help identify things that are going well, but they can also pinpoint potential blockers.

Common SRE best practices

Similar to Google's principles, PagerDuty has its own set of SRE principles and best practices that can serve as a model for your organization. These include:

- Knowledge sharing, so that SRE operational expertise is in service to the entire organization
- All teams following operational standards
- Partnering with engineering stakeholders to define a supportable service architecture
- Creating self-service capabilities that deliver automated and repeatable services on demand
- Providing opinionated defaults and guardrails that propagate service reliability
- Continuously improving the customer experience for internal development teams across the full lifecycle (creation, development, deployment, retirement), observability, flexible connectivity, and monitoring)

How does PagerDuty empower the SRE?

"Stream-aligned" is how we at PagerDuty define our service ownership model. It's about orienting the team around a capability or service, so that they are fully enabled, integrated, and capable to build, ship, and run.

The PagerDuty Operations Cloud is an essential infrastructure based on a service-based architecture. It automatically detects and diagnoses disruptive events from across digital operations, mobilizes the right team members, and automates infrastructure and workflows to resolve unplanned, time-sensitive issues.

You will find SREs supporting teams and services of all kinds throughout digital operations. In fact, customers may deploy multiple SRE models for the various functions, e.g., from NOC Operations, Platform SREs, to DevOps. What truly sets PagerDuty apart is how we enable and empower the various operating models, including ensuring proper alignment and visibility of service ownership. PagerDuty empowers SREs with self-service capabilities, such as:

- Creating and managing their support schedules
- Defining the required monitoring & telemetry on their services to optimize visibility
- AIOps capabilities to reduce noise, improve context and expedite detection and resolution
- Automation capabilities to reduce toil, drive auto diagnostics and self-healing
- Flexible workflows to ensure proper and immediate incident response while optimizing human engagement
- Deploying the data, operational metrics and trending analytics on services and teams to continually identify additional opportunities for automation and process optimization

Looking Down the Road

SREs, by nature of the role, often spend more time on stewardship and strategy. Some SRE groups help spread the message of quality and create the strategy of application production support for the organization. They collaborate with developers to better understand how SRE processes can provide more support. They choose key metrics, decide how automation will be set up and consumed, and create policies on how on-call should function. The list goes on.

In a sense, SREs are a kind of "Center of Excellence" for your organization—not just about the day to day but focusing on the bigger picture.

Future challenges and opportunities

The biggest challenge many organizations face is finding qualified SREs. "There's just not that many folks in the industry yet who are skilled enough and available enough," Walls said. "There's a lot less on-ramp for an SRE across the industry—there's not a place to go and say, oh, now that I know some code, I want to learn how to be an SRE. From that perspective there's sort of a hole in the market."

Walls said she expects to see more training over the next few years specific to onboarding SREs. She also believes there will be more focus on the actual user experience of applications, vs. CPU utilization or something similar. "There will be more of a drive toward better user experience and not just chasing down whatever metric is available," she said.

Digital transformation, flexibility and modernization are more essential and happening faster than ever. In this new paradigm shift, SREs play a crucial role in helping organizations align their goals for reliability to achieve robust digital experience that support overall business objectives.

To see how PagerDuty can empower your SRE Teams to service ownership model, visit www.pagerduty.com or start a 14-day free trial today.

PagerDuty, Inc. (NYSE:PD) is a leader in digital operations management. In an always-on world, organizations of all sizes trust PagerDuty to help them deliver a better digital experience to their customers, every time. Teams use PagerDuty to identify issues and opportunities in real time and bring together the right people to fix problems faster and prevent them in the future. Notable customers include Cisco, DocuSign, Doordash, Electronic Arts, Genentech, Shopify, Zoom and more.

To learn more and try PagerDuty for free, visit www.pagerduty.com. Follow our blog and connect with us on Twitter, LinkedIn, YouTube and Facebook.

2023 PagerDuty® // All Rights Reserved.

PagerDuty