BLAMELESS

SLO Maturity and SRE Adoption Report 2021

Table of Contents

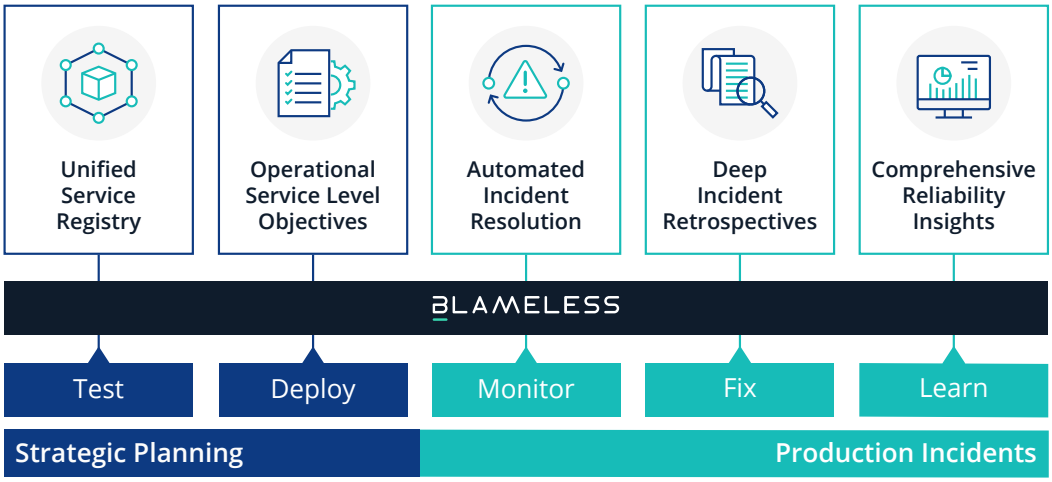
Introduction: Service level objectives and why they matter

Uptime is critical to organizations, and customer happiness more important than ever. To adapt, teams globally are adopting site reliability engineering (SRE). SRE is a growing trend, with job openings increasing by 72% YoY. The rise in job openings is no surprise. Organizations are looking to become more reliable. Customers need services to be available when they need it, and that can often be 24/7. In short, reliability has become feature No. 1.

One of the most foundational SRE best practices is service level objectives (SLOs). SLOs allow organizations and software teams to keep customer experience at the center of decision-making. In a nutshell, service level objectives are an internal goal for the essential metrics of a service. These essential metrics are distilled into SLIs (service level indicators). They state the level of service that customers will find acceptable.

The inverse of the SLO is the error budget, or the allowable amount of failure for a particular time window. Error budget burn is what informs decisions about reliability versus feature velocity. When the error budget as well burn rate are within limits, teams can push new code. When teams exceed the error budget or the burn rate accelerates, it's time to focus on reliability efforts.

SLOs are the foundation of a successful SRE practice. They help organizations make truly data-driven trade offs between speed and quality to deliver optimal user experiences. As they transform software teams' ability to focus on the work that matters most, we launched a survey to better understand the state of SLO maturity and adoption. The 285 survey respondents spanned many industries, organization sizes, and regions, and the results collectively reveal the measurable business impact of SLOs.



Executive Summary

Many organizations today are investing in maturing their operations practices. This allows them to create better alignment across the business. Yet they face challenges such as prioritizing between speed (planned work) vs. quality (unplanned work), maintaining legacy systems, and rapidly increasing microservices complexity.

As such, we're seeing the rise of practices such as SLO adoption. **80% of organizations indicated that they are already using SLOs, or plan on doing so within the next 1-2 years**.

SLO adoption has several tangible benefits for the business. In the survey, we grouped organizations into three key stages of maturity, which were defined as follows:

Beginner:
We have just started or are in the exploration phase

Intermediate:
We have a few services with SLOs

Advanced:
Some, most, or all of our services have SLOs

The survey found that organizations at an *Advanced* stage of SLO adoption are far more confident in their team's ability to prioritize sprint work, increase innovation velocity without impacting user happiness, understand and reduce technical debt. In fact, **55% of teams at the *Advanced* stage understand, discuss, and prioritize technical debt**. This is 2-3X greater than that of less advanced teams.

Additionally, *Advanced* teams have better cross-functional alignment across product owners, SRE & operations, software engineers/developers, and other business stakeholders. These less-siloed organizations are able to deliver greater value to customers faster.

Key Findings

Here are some key findings of the study, which focused on understanding the state of SRE and SLO adoption across organizations, top business goals and challenges, and the impact of SLOs on the business.

1 SRE is growing, with organizations of all sizes and maturity adopting SRE practices

- **Over 50% of teams we surveyed employ an SRE model** with dedicated engineers focused on infrastructure and tooling, or an embedded SRE model where full-time SREs are assigned to a service.
- **The most widely adopted SRE practices** include CI/CD (64.8%), Incident Response (59.5%), Observability (50.7%), and Postmortems or Incident Retrospectives (47.5%).

2 High interest as well as barriers to SLO adoption

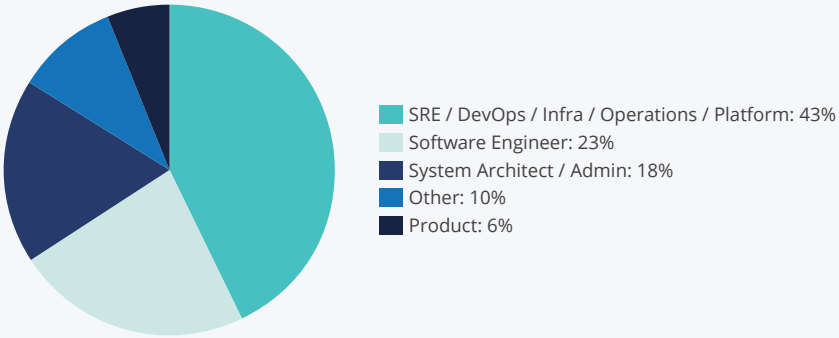
- **80% of organizations** indicated that they are already using SLOs, or plan on doing so within the next 1-2 years.
- **Most teams struggle with barriers to SLO adoption.** Top challenges were a lack of cross-functional processes to define and periodically review SLOs (45.8%), and data that is siloed across teams and tools (41%).

3 The measurable benefits of SLO on the business

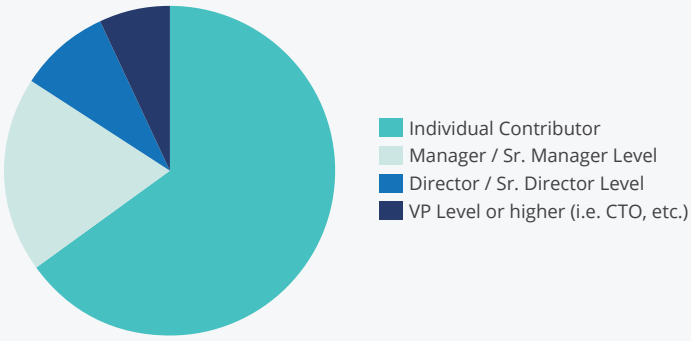
- **Advanced teams are 2-3X more likely** to understand, discuss, and prioritize technical debt across the organization.
- **Advanced teams are almost 3X more likely** than other teams to be very confident in increasing feature velocity without impacting user happiness.

Survey Methodology

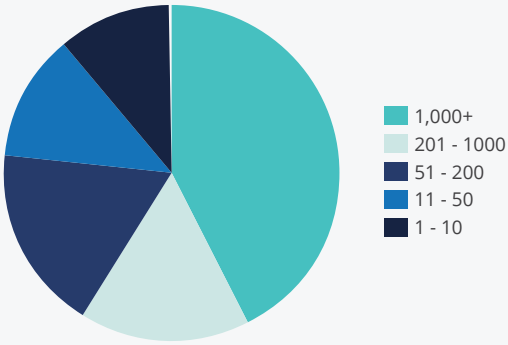
What is your job title?



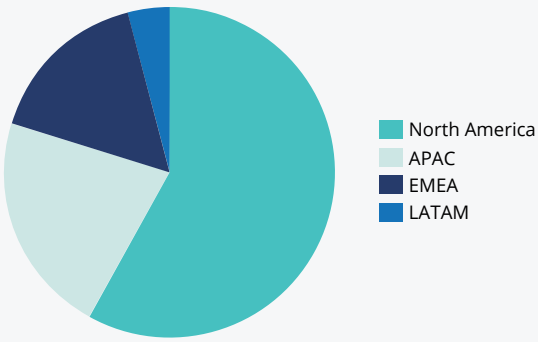
What seniority level are you?



Org Size



Location



SRE adoption is becoming mainstream

The practice of SRE is growing rapidly, with many organizations large and small adopting SRE best practices. One reason for the rise in SRE adoption is the increasingly complex and mission-critical challenge of protecting customer experiences. According to the survey, there are three major factors that increase the difficulty.

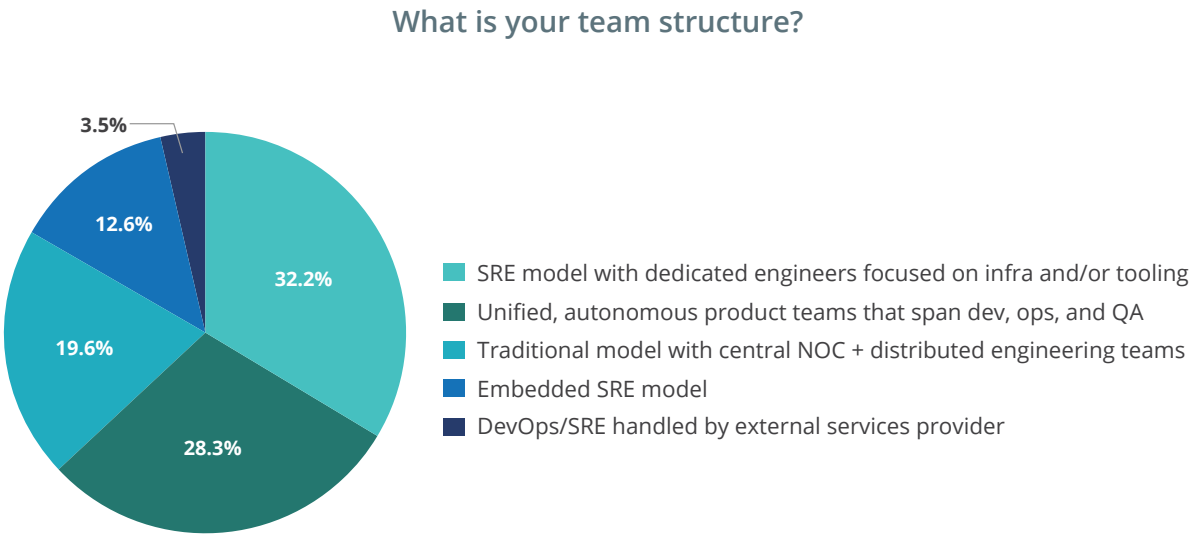
51% of respondents stated that their organizations face difficulty prioritizing speed versus quality. Traditionally, the goals of development and operations have been inherently misaligned, with developers oriented around speed and operations oriented around stability. This can cause a lack of alignment on goals and thrash between teams. SRE, and in particular SLOs, provide a methodical way to prioritize planned work in a way that balances these competing priorities and thereby optimizes user experience.

Another top challenge in protecting customer experience is increasing operational complexity. **Over 47% of respondents said this was a major concern.** With the increase of microservices, each change has a ripple effect throughout the system.

SRE can help ease the burden of operational complexity in several ways. Incident response best practices can help teams get their services running as expected, faster with tools like runbooks, incident management automation, checklists, and more. After incidents, the retrospective (or postmortem) can serve as a learning tool, allowing teams to glean insights into complex systems and avoid repeat incidents.

Respondents (41%) also listed maintaining legacy systems as a top challenge. Legacy systems can be fragile, and difficult to make changes to. Best practices like CI/CD can help teams roll out new code at a faster clip. Thorough documentation, runbooks, and automation drive predictability and speed at every stage of the software lifecycle. Additionally, SLOs help teams prioritize when to push new code, and when to burn down the technical debt.

SRE is able to assist with some of the thorniest problems that teams face. But, there is more than one way to operationalize SRE within an organization. Team structure can vary, as demonstrated in the chart below.



Current operational practices

Adopting SRE best practices has a marked impact on organizations' ability to deliver great software while cultivating well-aligned, high-performing engineering organizations. But, teams are adopting key SRE practices to varying degrees.

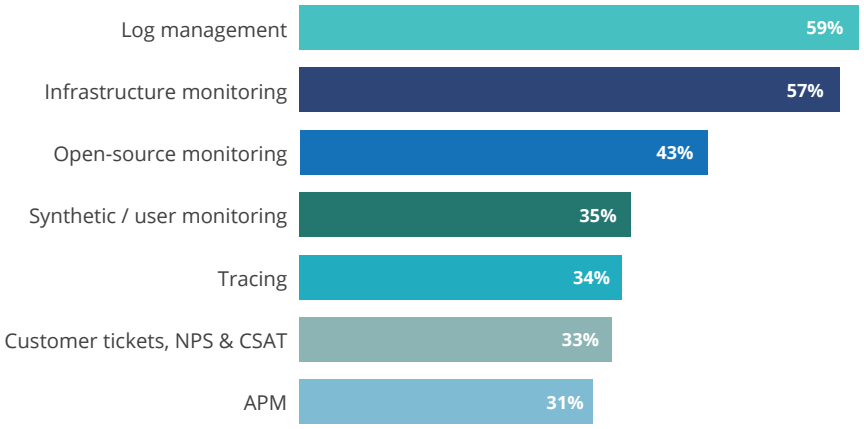
Which practices does your team engage in today?

Survey respondents reported fairly wide adoption of best practices such as: CI/CD (64.8%), Incident Response (59.5%), Observability (50.7%), Postmortems or incident retrospectives (47.5%), and Capacity planning (43%).

SLO and service cataloging are somewhat more nascent practices, at around 30% adoption respectively. This is likely due to the complexity as well as breadth of stakeholders required to effectively implement these practices, especially in the context of distributed systems and microservices.

Today, many organizations are using multiple methods to understand end users' experience, and operational data is often fragmented across many different types of monitoring and observability.

Which systems are you using today to understand end users' software experience?

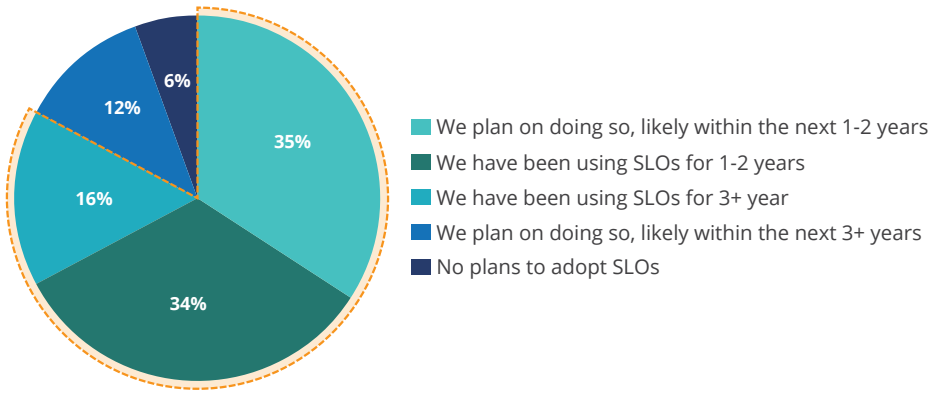


With the explosion of tools and methodologies to gain observability into complex systems, it becomes increasingly hard to prioritize and understand how to parse the signal from the noise. SLOs are highly valuable in rationalizing this data and driving focus, by aligning teams around shared definitions of user journeys and parameters of user happiness. As such, as SRE practices such as observability become increasingly mainstream, we see that SLOs are becoming widely adopted as well.

Top reasons for interest in SLO adoption

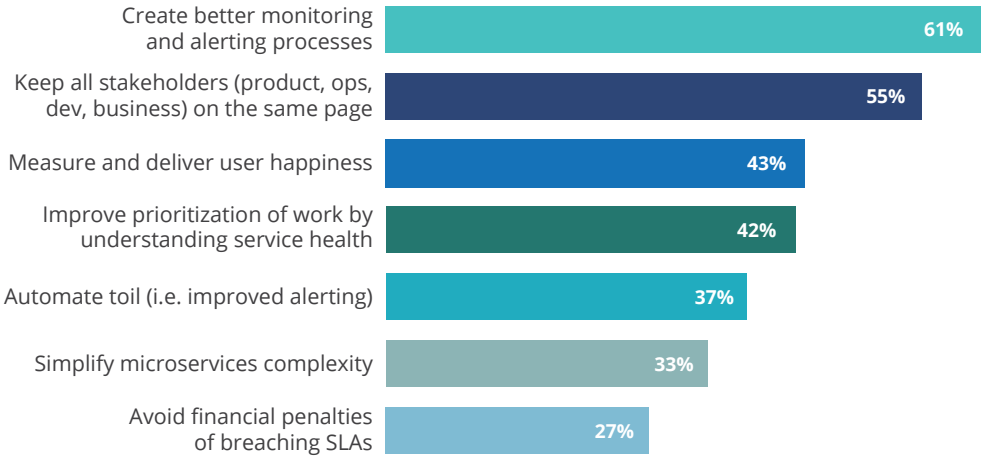
According to the survey, **over 80% of organizations already have adopted SLOs, or plan on doing so within the next 1-2 years**.

When does your organization intend to adopt SLOs?



Respondents reported several key areas of interest driving SLO adoption, as shown in the chart below.

What are your organizations main interests in SLOs?

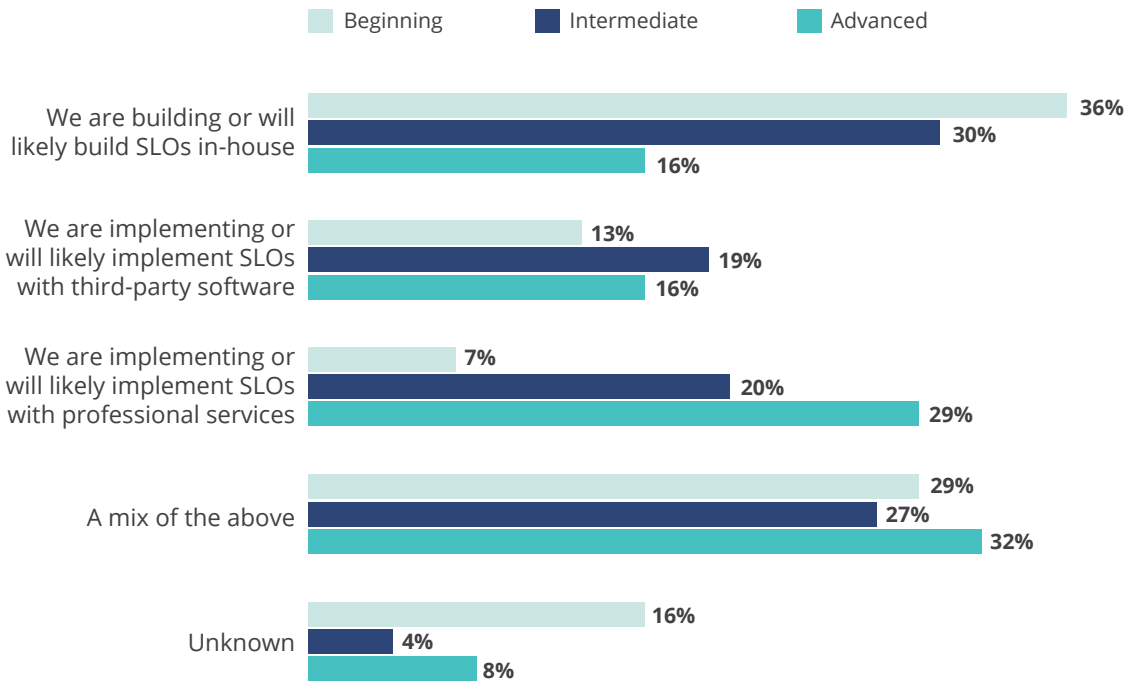


The most common reason was for improved monitoring and alerting processes (61%). Traditional alerting can often result in higher levels of noise and alert fatigue, and even 'AIOps' approaches often comprise black box machine learning, or rules and thresholds that can quickly become stale in context of complex systems. While SLOs also are never 'one-and-done' and require ongoing tuning over time to truly be effective, they are much more accurate proxies for the end user experience . Creating alerting thresholds based on error budget burn also helps teams gain a better idea of what's important to alert on.

Another top interest in SLOs was to improve stakeholder alignment around the customer experience (55%). Often, different teams can have different motivations. Operations wants to limit changes and protect the reliability of the system. Development wants to create code. Product wants to ship new features to delight customers. Leadership wants to increase profits and customer satisfaction. When these priorities are at odds, SLOs can act as a powerful mediation tool. **They help all stakeholders align on what's most important to the customer *and* the system.**

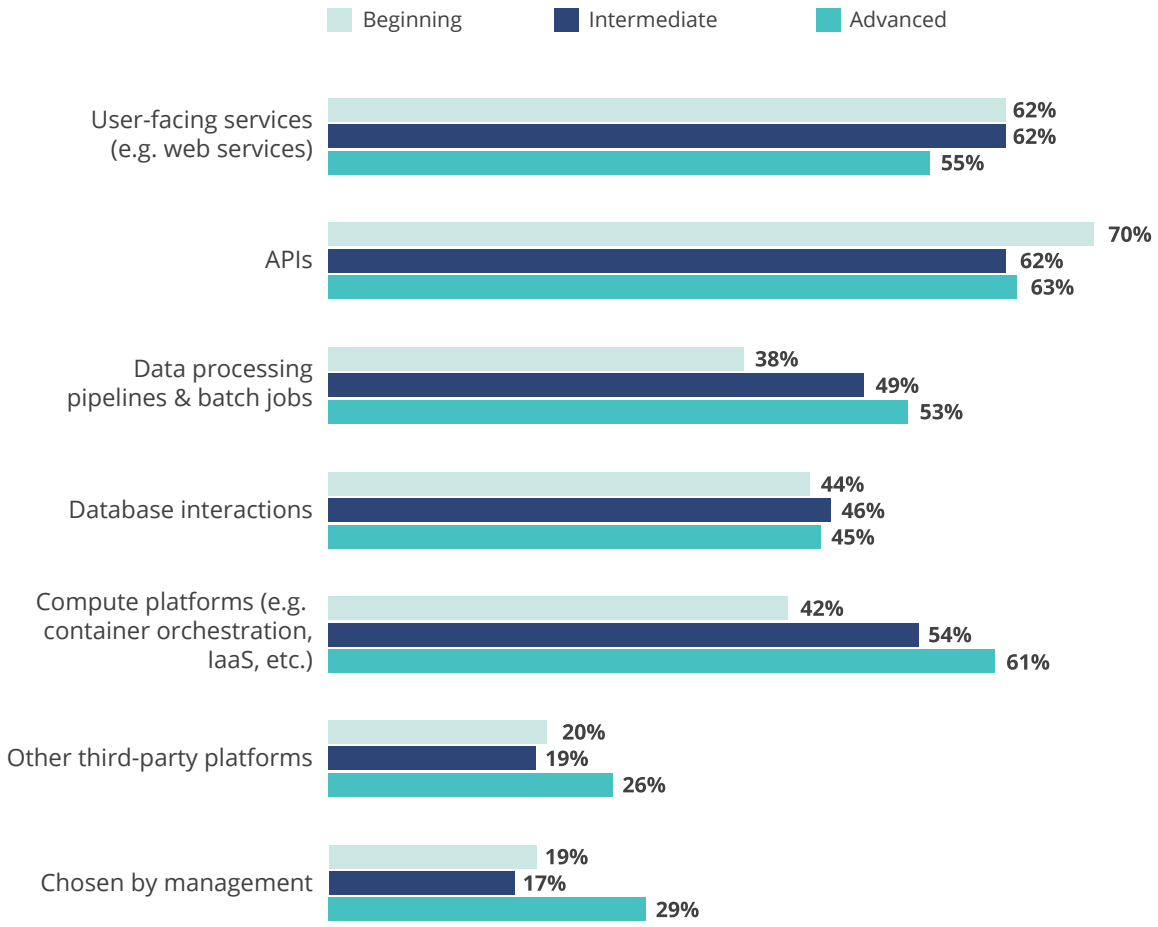
Teams are also approaching SLO implementation in various ways, which include building them in-house, leveraging software or tooling, consulting a third party, or a combination. We also see that approaches tend to vary across organizations of different SLO maturity levels, where *advanced* teams are most likely to use a combination of approaches together, while less *advanced* teams are more likely to approach SLO implementation in-house.

What is your team's approach for SLO implementation (now or in the future)?



Teams at different stages of SLO maturity have embraced different methods to implement SLOs. However, across the board, the types of services teams want to measure with SLOs are quite similar**. The majority of organizations are using SLOs to measure APIs (64%) and customer-facing services (59%).** Over 40% report also setting SLOs for compute platforms, database interactions, and data processing pipelines & batch jobs. Currently only a limited percentage (19%) are chosen by management, suggesting that SLO projects may often originate as bottoms-up instead of top-down initiatives.

What types of services do you measure, or are interested in measuring, with SLOs?

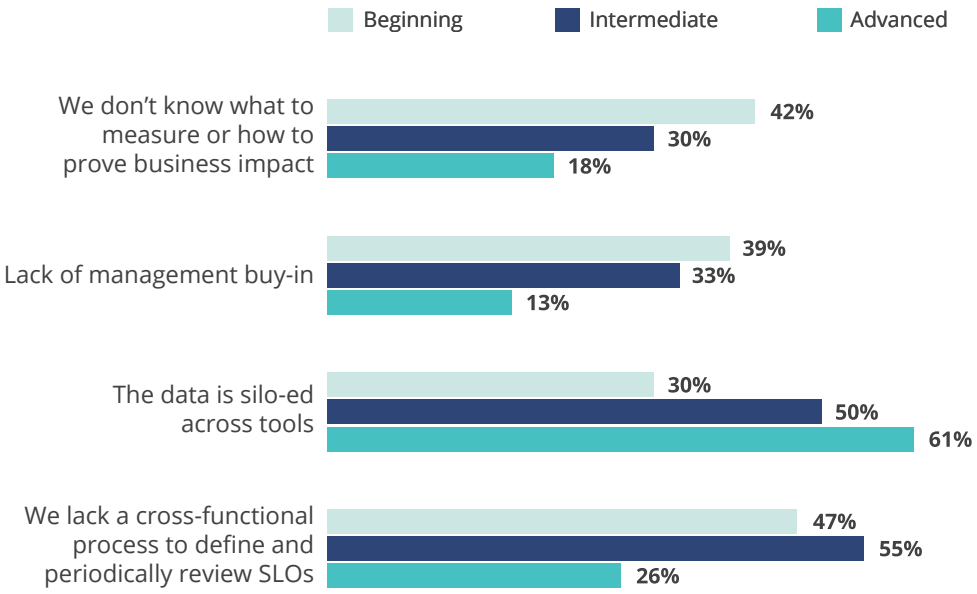


Yet, while the *why* and *how* of SLO adoption are now clear, we see that there are often significant barriers to overcome.

Barriers to SLO adoption

Successful SLO adoption isn't an easy process. It requires significant stakeholders, data, tools, time, and resources. Our respondents noted the most difficult challenges they face during the adoption process.

What have been the primary challenges/barriers to SLO implementation or adoption?



For *Advanced* teams, the greatest barrier to SLO adoption is that the data is silo-ed across tools. More mature teams typically often have more tooling and observability in place, which means that they need to aggregate and contextualize data across a variety of sources.

Less advanced teams' greatest barrier is that they lack a cross-functional process to define and periodically review SLOs. There are several possible explanations for this. First, it can be difficult to communicate the importance of SLOs to all stakeholders and gain buy-in.

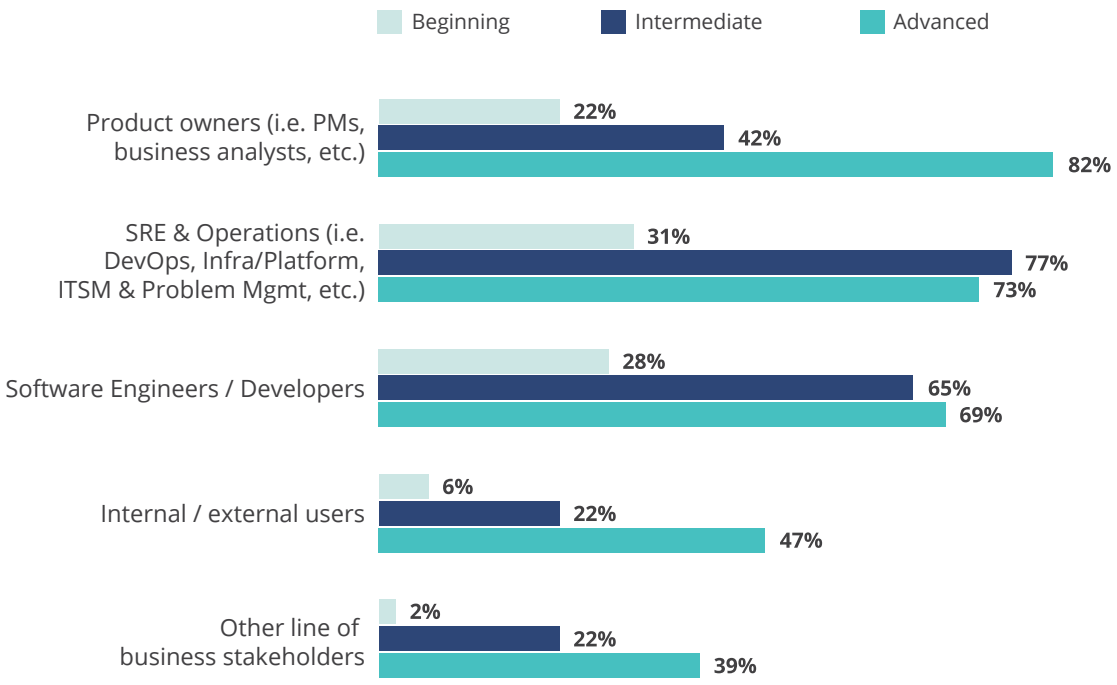
Additionally, a lack of internal processes can also hinder SLO adoption. SLOs are not set-it-and-forget-it tools. They need to adjust as customer expectations change. They will also need to scale with your business needs. Teams will need to determine a cadence to review and iterate on SLOs on an ongoing basis.

Characteristics of *Advanced* organizations

According to the survey, organizations which are more advanced in their SLO adoption often share certain characteristics. For example, *Advanced* teams in the survey tended to be larger. **63% of *Advanced* organizations that responded as advanced in their SLO maturity also had 1000+ people in their organization.** These organizations often have the resources to scale SRE. They also likely have larger customer bases, meaning a larger impact when incidents occur. Uptime is critical for all organizations, but organizations with millions of users have very high stakes.

No matter what the organization size, *Advanced* teams tend to have better cross-functional engagement in the SLO process. Most *Advanced* teams invite development, SRE & operations, as well as product to SLO discussions. This is in contrast to teams in the *Beginning* phase, who are significantly less likely to involve other teams. For instance, while 82% of *Advanced* teams involve product in SLO decisions, only 22% of *Beginning* teams do the same.

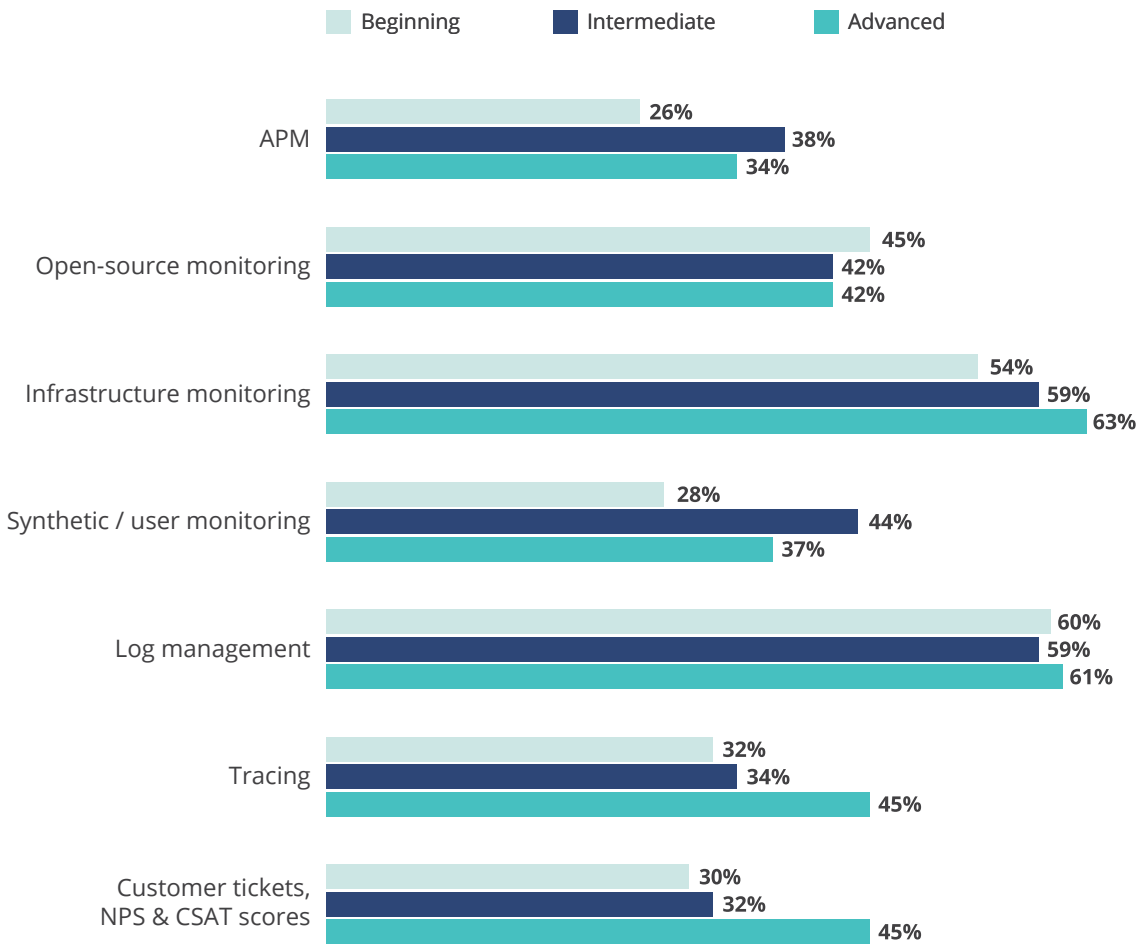
Who are the current stakeholders involved in the SLO process?



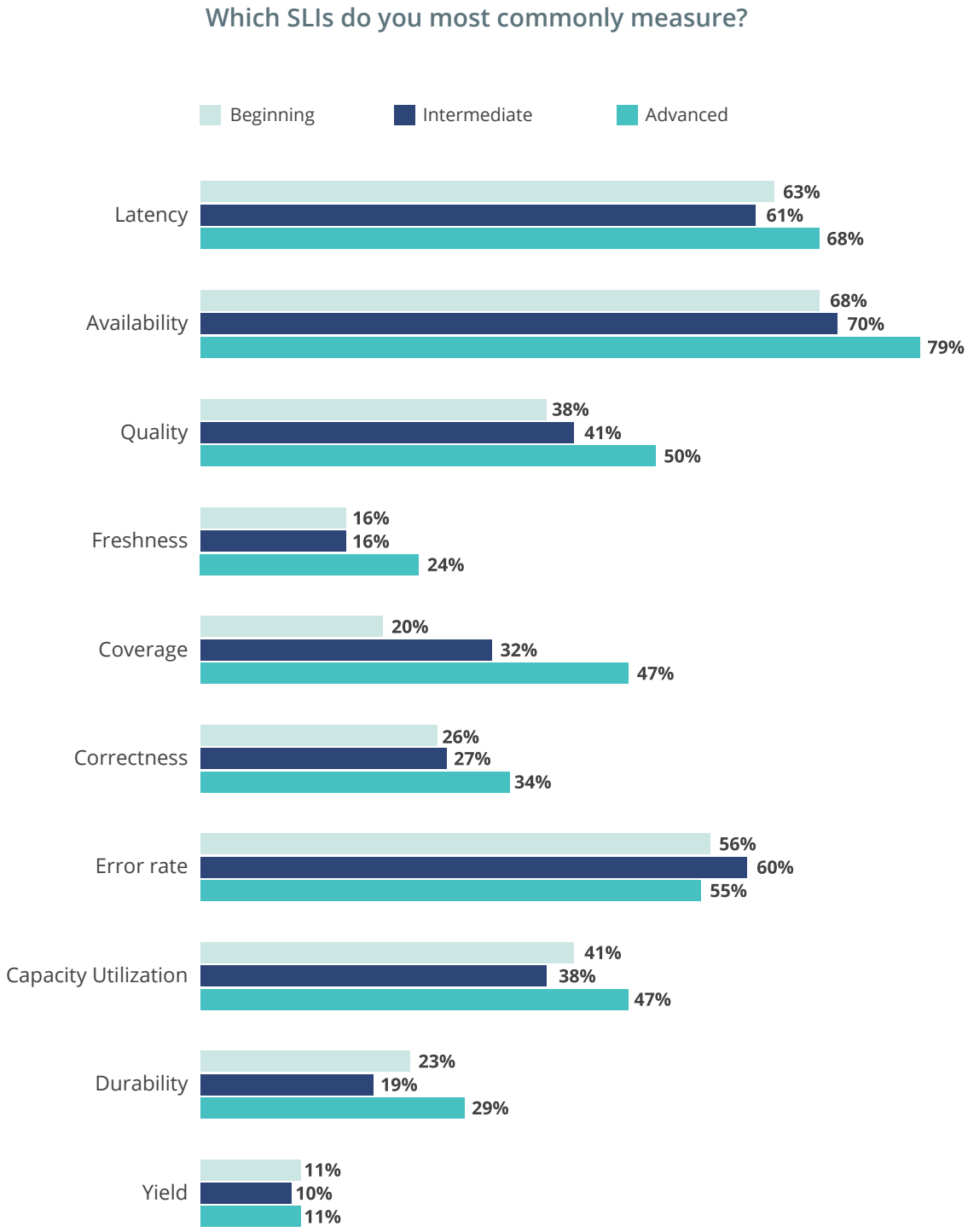
While teams of all sizes will likely experience bumps in the road on the journey to SLO adoption, there are significant, quantifiable business benefits that make the investment worthwhile.

Advanced teams also tend to have better observability, with greater coverage across a much wider swathe of tools as well as SLIs and metric types. For example, *Advanced* teams use more *systems* to understand customer experience. *Beginning* and *Intermediate* teams primarily evaluate customer experience by infrastructure monitoring and log management. While *Advanced* teams also use these systems, they also augment with tracing, customer tickets, and NPS scores. Direct customer input (customer tickets and NPS scores) are the largest delta. While less than a third of *Beginning* and *Intermediate* teams use these metrics, almost half of *Advanced* teams do.

Which systems are you using today to understand end users' software experience?

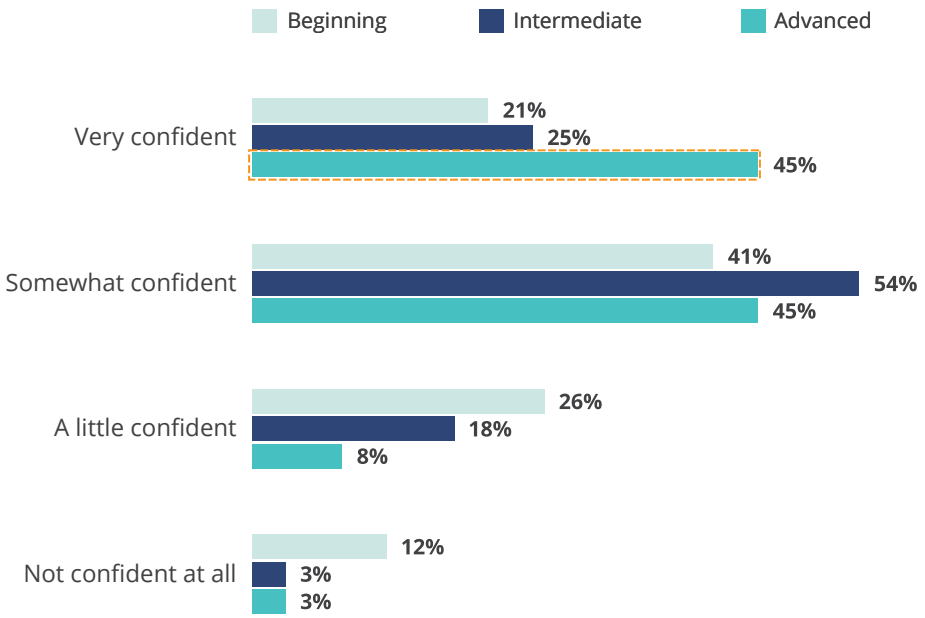


Advanced teams also monitor more categories of SLIs. **While *Beginning* and *Intermediate* teams typically only measure latency, availability, and error rate, *Advanced* teams have significantly wider breadth across different SLI types.** This allows them *Advanced* teams to more effectively uncover points in the product where users stub their toes. Delving into these points on the user journey helps increase customer happiness and lower churn.

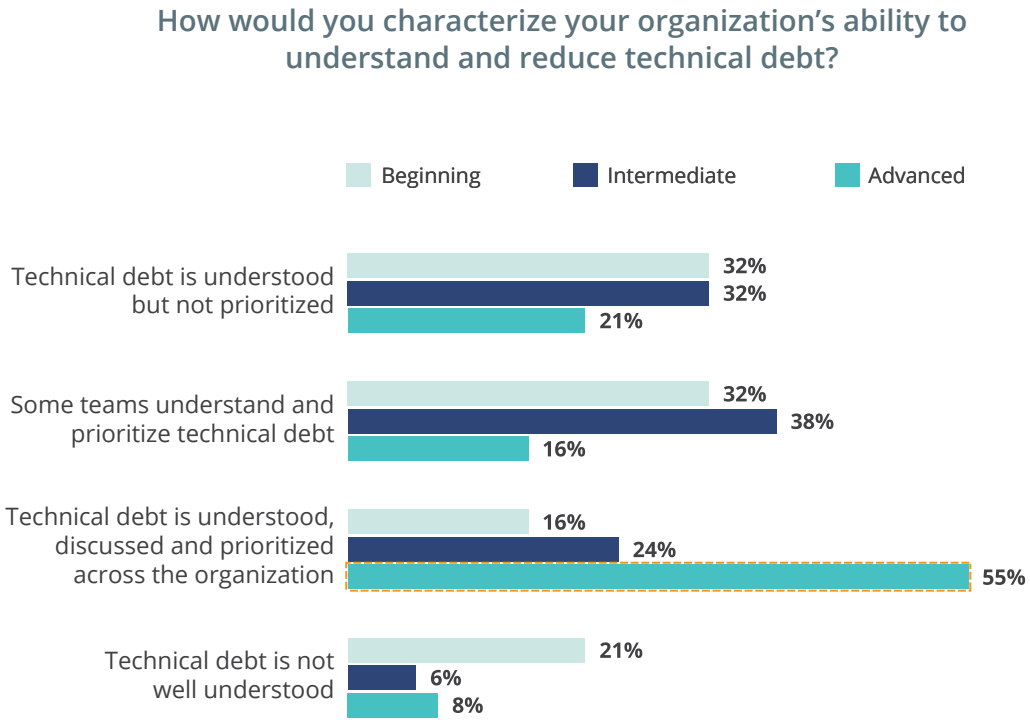


Advanced teams not only have greater metrics coverage and visibility, but are also measurably more confident in cracking the code on the archetypal 'speed-reliability' compromise. 90% of *Advanced* teams are either "Very Confident' or 'Somewhat Confident' in prioritizing between project work versus operational work, and are 2X more likely to be 'Very Confident' than less advanced teams.

How confident are you in your organization's ability to protect customer experience by prioritizing between project work (speed) vs. operational work (reliability)?



This increased confidence in balancing tradeoffs between project work (planned) vs. operational work (unplanned) is likely connected to other benefits, such as better management of technical debt as well as greater speed and confidence in deployments. For example, *Advanced* teams are **2-3X more capable than *Beginning* and *Intermediate* teams in understanding, discussing, and prioritizing technical debt across the organization**. As SLOs shape the prioritization of planned work based on guidance from error budget policies, teams are empowered to not just assess but fix technical debt by having a more quantifiable way to analyze and reason about its impact on users.



Advanced teams were almost **3X more likely to be 'Very Confident' in increasing feature velocity without impacting user happiness**, compared to less advanced teams. This can be due to a sense of engineering ownership as teams operate the code they write (service ownership model). This can also be attributed to a sense of confidence in the release process. *Advanced* teams release often, have detailed production readiness checklists, and have detailed procedures for recovery in the event of a failure.

Conclusion

The results of the survey reveal that SRE and SLO adoption have already become mainstream, and will continue to increase over the next few years. Teams are realizing the benefits of SRE and SLO adoption, which has meaningful and measurable impacts on the business.

This report reveals that the most challenging parts of effective SLO implementation isn't only about collecting data. Rather, it's about removing data silos, getting executive and stakeholder buy-in, and creating a cross-functional review process to operationalize SLOs. Additionally, it's important that teams work to adopt more than just SLOs for a robust SRE implementation. Other best practices such as CI/CD, incident response, incident retrospectives, and more help bolster SRE practices.

Finally, teams looking to operationalize SLOs need to consider both the tools available to them during the process, as well as the stakeholders needed to make SLOs a success. Teams who involve all stakeholders up front will be more successful in the long-run.

Blameless provides a vendor-agnostic approach to help operationalize SLO adoption, as well as tie SLOs to other core operational practices such as incidents, retrospectives, and more. To learn more, reach out to our team for a demo.