42 LINES

Demand More from Your Technology.

It's time to measure what matters.

In our experience helping companies introduce DevOps and modernize their approach to technology, we've seen methodologies come and go, from Waterfall to Scrum, from Agile to Lean, and more. These practices, mostly successfully, delivered better software faster. Today, the practice of DevOps brings together software methodologies with operations rigor, breaking down silos to create teams who are better equipped to keep pace with digital innovation, scale operations more successfully, and compete more aggressively in a business environment where the barriers to entry have fallen dramatically.

Different ways of working requires a cultural shift. DevOps in particular has blurred the lines between software development and operations in the push to automate tasks in order to deliver more value to the business.

On the one hand, bringing teams closer together to deliver better software faster has been a game-changer, improving collaboration and integrating processes across software development and operations.

On the other hand, traditional dev teams and ops teams have different goals, different vocabularies, even different incentives for getting things done. Dev teams want to launch new products. Ops teams want to make sure things don't break. One team (dev) is playing offense. The other team (ops) is playing defense. To complicate matters, dev teams themselves often instrument their code differently from each other. And that makes it difficult for an engineer (or anyone, really) to move between teams and easily understand what's going on.

How, then, do we bring these teams together in a way that delivers real business value?

Explore Our Upcoming and On Demand Webinars

42lines.net/resources/webinars

FOLLOW US

We believe this question is critical for companies to address, especially if they are embarking on a modernization effort. Metrics are useful, but we also want a generalized way of sharing knowledge.

Establishing metrics to demonstrate how well your technology services are working and what results they are achieving is at the heart of Site Reliability Engineering, a set of common practices that helps DevOps teams bridge the gap not just between development and operations, but between DevOps and the business. A discipline advocated by Google over the last five years, SRE codifies the practices of the DevOps team into meaningful measurements.

Demand More from Your Technology.

It's time to measure what matters.

If your company is thinking of building a DevOps organization or if it's considering external DevOps expertise, ask yourselves these questions:

› What do you want to accomplish?

What will modernization look like at your business? Are you embarking on a mission to integrate applications within a cloud platform? Why? Who will benefit? How will they benefit? What does this transformation mean for your business?

› What will you measure?

Businesses today are awash in forensic data. But while observability metrics across the technology stack have improved, the value of these efforts to the business remains stubbornly difficult to measure. And that's because most don't have a method for transforming that information into a usable set of metrics to measure success. Determine what matters to your business. Will you measure how well your technology is serving your users? Will you measure how well it is serving your customers? What measures will you put in place to determine success?

› How will you communicate value?

How will you determine if your effort has been successful? Meeting project milestones is great, but what business metrics will you report to your CFO? Are you regularly releasing more stable software more consistently? Are users adopting the tools you've delivered? How does improved user adoption reduce costs, improve efficiency, or drive revenue?

One of the more useful tools introduced through Site Reliability Engineering is Service Level Objectives. In our experience, determining SLOs is a particularly useful way to codify a desired goal and measure how well that goal is being met or surpassed. How, for example, will you define meaningful availability? How many nines will you really need? Because SLOs document expected behaviors (how fast a user expects a web page to render, for example), their formulation brings dev, ops, management, and even customers to the table to share their various needs and expectations to achieve a balanced and achievable goal.

The same technique can also measure risk by allowing us to answer how close we are to violating the SLO. In a world where users (and that's all of us) want both new features and a stable experience, SLOs can help drive the business decisions that determine how to balance these needs. When services across the business are measured with SLOs, everyone--from engineers to the tech on call to the C-suite--has a shared understanding of what those services should deliver.

The principles of Site Reliability Engineering are giving DevOps teams the procedures and techniques they need to make a real difference, helping them deploy standard metrics against various services to understand how those services are performing. By measuring what matters, businesses are making smarter decisions and delivering better customer experiences. That's a powerful tool for achieving even the most aggressive business objectives.

Interested in learning how to take the first steps in implementing SRE in your organization? Establish realistic baselines and ensure the foundation is in place to grow your practice and avoid the traps of implementations that don't establish a careful balance of responsibilities.

Start Your Journey Today ➔