



DevOps visibility and metrics for driving business value

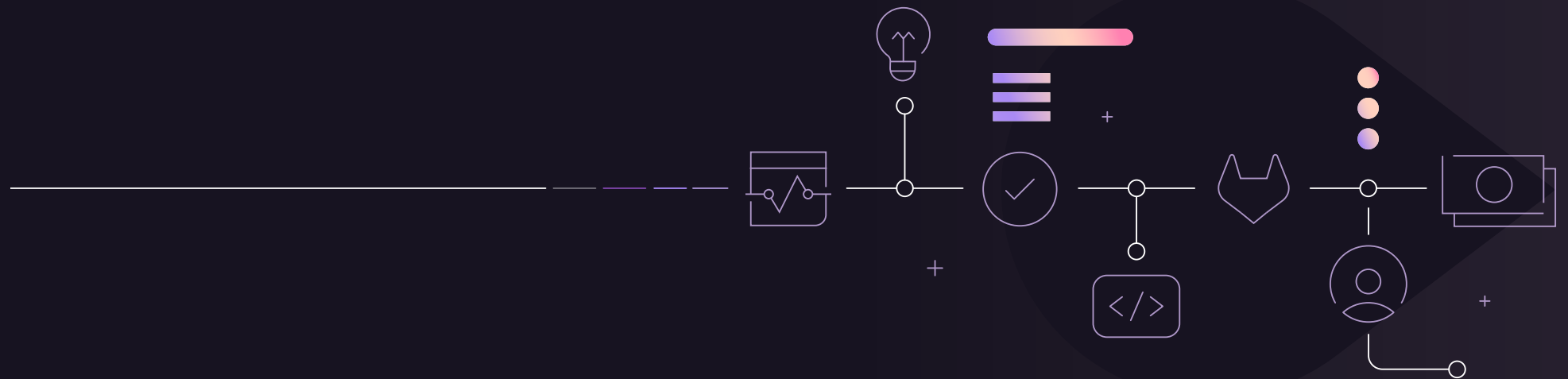


Table of contents

03 Introduction

04 What factors drive business value?

- Continuous improvement
- Quality output
- Cost effectiveness
- Organizational culture

05 Business value metrics for software development

06 Effectively measuring the value of software delivery

1. Visualize the end-to-end software delivery workflow with Value Stream Management
2. Tailor metrics to team needs with custom value streams
3. Measure software and team performance with DORA metrics

09 A single platform for business value insights

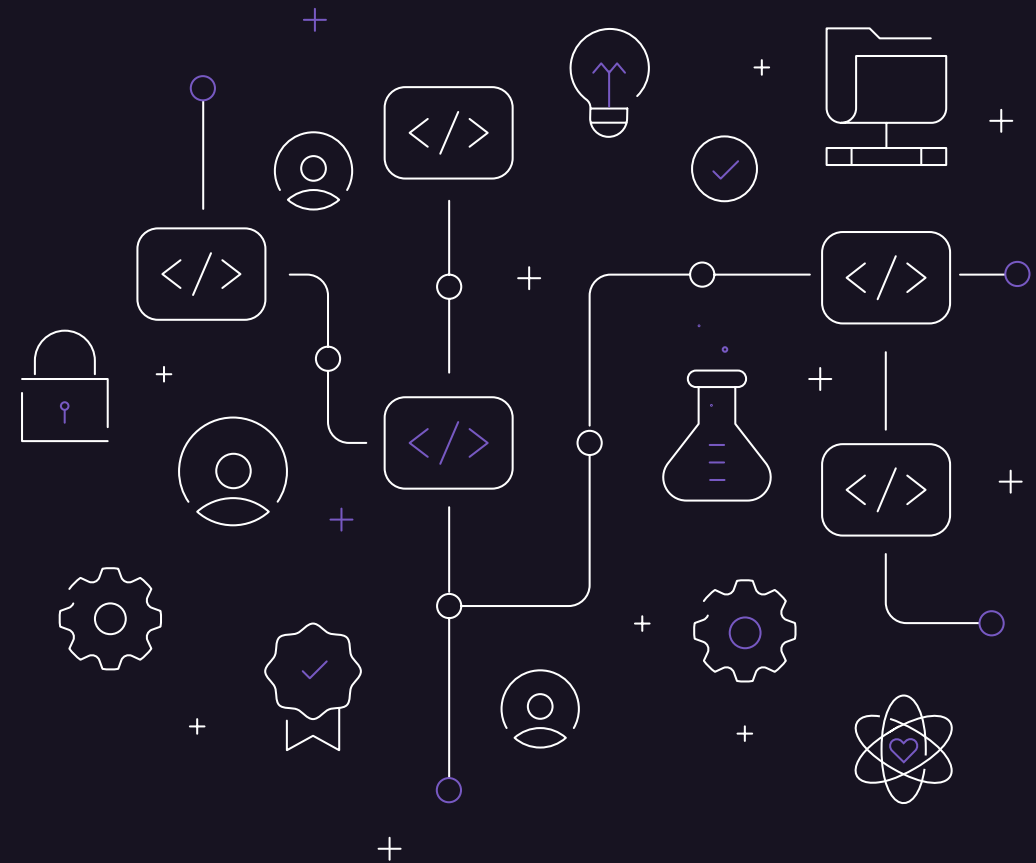
10 About GitLab



Introduction

Every company today must become a software company. To be successful, businesses must provide a high-quality digital experience, keep up with the latest technologies, deliver value at the speed customers demand, and do it all with zero tolerance for outages or security breaches. Improving the performance and quality of software is a top priority for many businesses, but this has been a challenge for most organizations. What aspects of the software delivery lifecycle are moving the needle, and how can teams better understand the impact their work is having on the overall business?

To reliably measure the business value of the software development process, organizations need better visibility across the software supply chain and a deeper understanding of business metrics. This guide provides an overview of the factors that drive business value for software development teams — continuous improvement, quality of output, cost effectiveness, and organizational culture — as well as specific metrics organizations can track, and how to effectively measure the value of software delivery.



What factors drive business value?

Application development teams are typically focused on improving the efficiency and quality of software delivery and creating repeatable processes — all with the goal of delivering products that meet the demands of customers and business stakeholders. But where specifically can organizations focus to help DevOps teams drive business value? Let's start with the factors that drive value for software development: **continuous improvement**, **quality output**, **cost effectiveness**, and the **extent to which the organizational culture is value-driven**, or geared towards value-driven processes and measurements.

Continuous improvement

The speed of delivery ultimately depends on the most time-consuming step in the end-to-end process: improve that one part and the whole process gets faster. **Continuous improvement** — also known as *Kaizen*, after the Japanese word for “improvement” — is a strategy that refers to making small, iterative changes that have major impacts. When applied in a business setting, continuous improvement has a significant effect on culture, productivity, and quality.

The classic continuous improvement cycle consists of **plan, do, check, adjust** (PDCA): identifying a change required to achieve a desired result, carrying out that change, evaluating data and results to confirm that the change is having the desired effect, and finally identifying areas where the process can be improved for the next iteration. Teams that implement this cycle are positioned to focus their efforts on one factor at a time, innovate more quickly, and take an objective look at what's slowing them down.

Establishing continuous improvement isn't necessarily difficult. But the complexity of **fragmented toolchains** and processes, **siloed incentives**, and lack of collaboration often get in the way of making lasting improvements in software delivery.

Quality output

Focusing on code quality is a great first step towards continuous improvement. An application's source code is a repository of knowledge, history, and solutions, so it's important to maintain the quality of the source code and protect it from harm. Implementing code reviews — systematic assessments designed to identify bugs — helps to ensure the quality of the code base and prevent unstable code from shipping to customers. Code reviews help organizations share knowledge within and across teams, identify bugs more quickly, maintain compliance, and improve security.

Despite the obvious benefits of encouraging developers to have their code reviewed early and often, **adding a code review step can delay the release process and pull focus from other tasks**. The solution lies in code review tools that include automated testing and comprehensive version control.



Cost effectiveness

In the face of increased IT demand due to the growing prevalence of remote work — plus external financial pressures such as inflation, supply chain issues, and currency volatility — spending IT budgets efficiently could be a key factor in success for many organizations.

In the context of software development, that could mean consolidating complex DevOps toolchains, eliminating clunky and expensive legacy systems, automating manual tasks, and weeding out inefficient processes. Organizations that can reduce costs while continuously improving the quality of their output are best positioned to remain at the top of the pack, but maintaining that balance may require teams to rethink existing tools and management processes. In [GitLab's 2023 DevSecOps Survey](#), 90% of respondents said they use a DevOps platform or are considering adopting one in the next 3 years. Developers also identified cost and time savings and a more streamlined DevOps practice as the key benefits of a platform approach.

Organizational culture

After establishing a cycle of continuous improvement, sharpening focus on quality, and eliminating waste and inefficiencies to make the most of software development budgets, the last aspect to driving business value is actually measuring that value: [knowing the right things to measure](#) and building a data-driven DevOps culture within the organization.

Even after identifying a handful of metrics, organizations may soon find that comparing data between different development groups can be a challenge. It's important to adopt a set of metrics that allow for standardization between teams and benchmarking against the industry.

Business value metrics for software development

A metric of business value could be any quantifiable measure that can be used to judge how a business is performing. Because there are many different types of businesses, there are likely hundreds — if not thousands — of possible business metrics, from sales revenue to lead conversion rate to talent retention.

In the context of software delivery performance, [DevOps Research and Assessment](#) (DORA) has created a list of four metrics that are straightforward and focused. They form a foundation for metrics initiatives that can help to improve existing DevSecOps efficiency while also potentially building a bridge to business stakeholders. The four DORA metrics are:

- **Deployment frequency:** the rate of successful deployments to production (hourly, daily, weekly, monthly, or yearly)
- **Lead time:** how long it takes to deliver a feature once it has been developed
- **Mean time to recover:** how long it takes an organization to recover from a failure in production
- **Change failure rate:** the percentage of deployments that cause a failure in production



Effectively measuring the value of software delivery

The ultimate goal for many Agile and DevOps teams is to demonstrate the business value created by software delivery. Understanding the metrics that matter to an organization is essential, but teams also need to effectively measure those metrics to know they're on track. Here are three steps teams can take to effectively measure the value of software delivery:

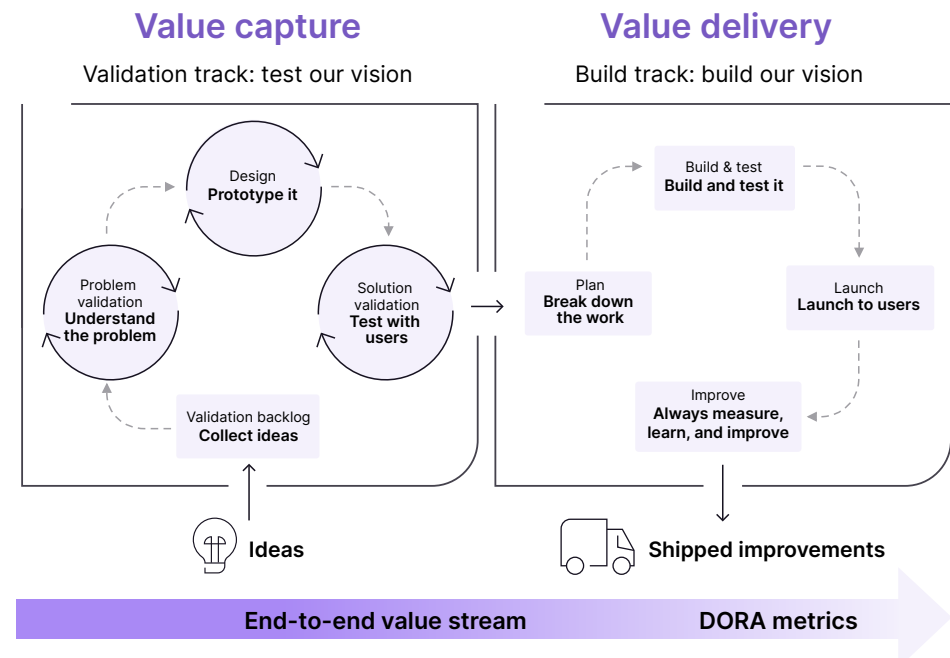
1. Visualize the end-to-end software delivery workflow with Value Stream Management

To keep up with customer demands, businesses need to continually drive value improvements at scale. That's where **Value Stream Management** comes into play.

A **value stream** is the entire work process that delivers value to customers. By monitoring and managing value streams — instead of features and functionality — throughout the software development lifecycle, organizations can measure the value of software delivery efforts and optimize the impact of their activities across the DevOps lifecycle. Value Stream Management can help organizations get a wider view across the software supply chain, helping to eliminate short-term delivery bottlenecks without losing focus on the long-term goal of delivering value.

Value Stream Management can help organizations identify whether DevOps teams are continuously improving their processes, flag potential issues with the quality of output, monitor cost effectiveness, and build a culture geared towards value-driven processes and measurements — all of which can drive value for the overall business. But, as mentioned above, fragmented toolchains and manual processes make this difficult work. With **Value Stream Management as part of GitLab's DevSecOps**

Platform, organizations can get out-of-the-box visibility into the end-to-end software delivery workflow. This can enable teams and managers to identify inefficiencies and take actions to improve team satisfaction and optimize workflows — all in one application, without the **DIY toolchain tax**.



2. Tailor metrics to team needs with custom value streams

Collecting the right data to make decisions is crucial. Metrics that are relevant to one team or a specific stage of the DevOps lifecycle may be less impactful in other contexts. Teams may want to use value stream analytics to identify:

- The amount of time it takes to go from an idea to production
- The velocity of a given project
- Bottlenecks in the development process
- Long-running issues or merge requests
- Factors that are slowing down the software development lifecycle

Customizable value streams in GitLab can help teams to monitor specific workflows tailored to their particular needs and identify high-priority blockers to delivering value to customers. With just a few clicks, users can identify a merge request stuck in code review or an issue waiting for approval and then quickly solve the problem, without wasting cycles on questions and clarifications.

Edit value stream

Start event

Issue label was added

Issue first added to a board

Issue first associated with a milestone

Issue first mentioned in a commit

Issue label was added

Issue label was removed

Merge request closed

Merge request created

Merge request first commit time

Merge request first deployed

Start event label

workflow::design

End event label

workflow::planning breakdown

Start event label

workflow::planning breakdown

End event label

workflow::in dev

Stage 4

Development

Start event

Issue label was added

Start event label

workflow::in dev

End event

Issue label was added

End event label

workflow::in review

Stage 5

Review (MR)

Start event

Merge request created

End event

Merge request first deployed to produ...

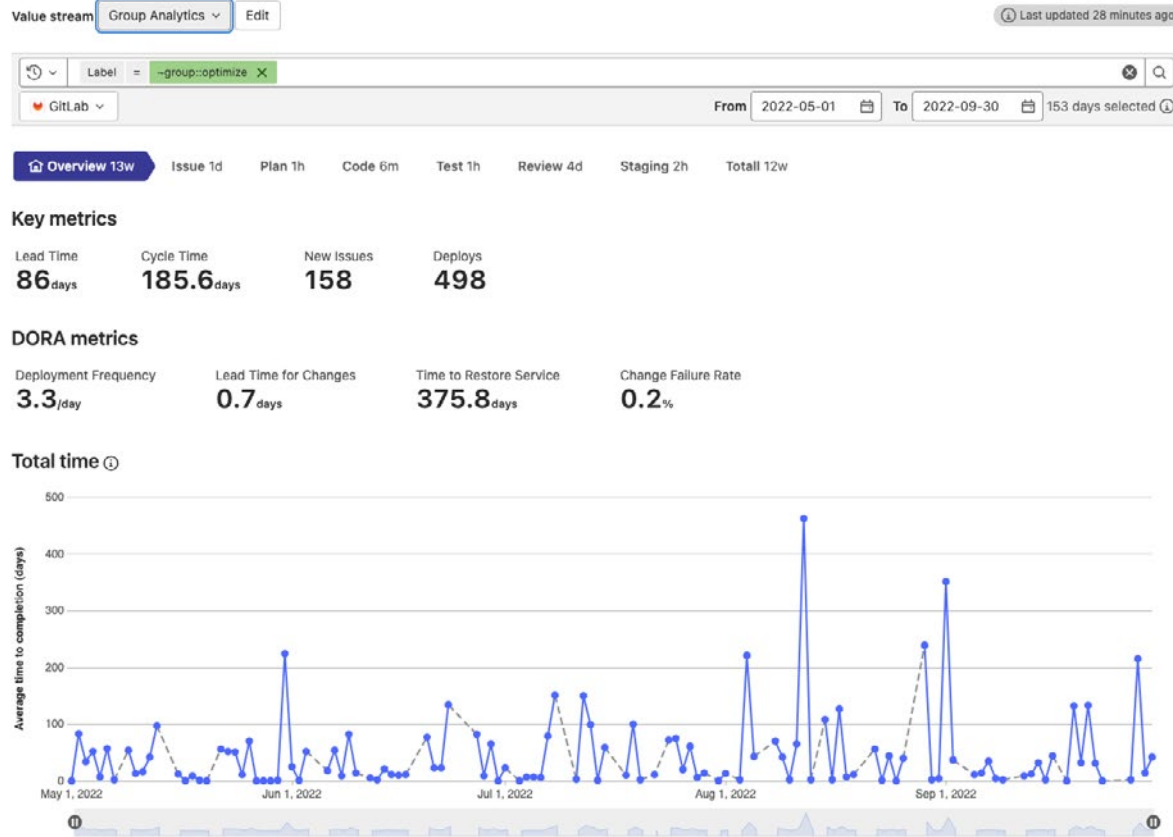
Cancel

Add another stage

Save value stream



Value Stream Analytics



3. Measure software and team performance with DORA metrics

To measure software delivery performance is to measure team performance — both are driven by velocity and impact. Measuring and optimizing DevOps practices can improve developer efficiency, overall team performance, and business outcomes. **And because the DORA metrics deliver visibility across the value chain**, tracking and optimizing these metrics can help teams streamline alignment with business objectives, drive software velocity, and promote a collaborative culture.

When U.K. real estate company **Zoopa** set out to measure improvements in the performance of their engineering department, they landed on the DORA metrics, which provided them with all the necessary insights to track success and benchmark against a definition of “good.” And **Zoopa chose the GitLab DevOps platform** to obtain metrics for deployment frequency, lead time, change fail rate, and time to onboard. Since they began to track DORA metrics and make them available to internal teams for decision making and prioritization, Zoopa has been able to incorporate changes into their planning cycles to stay on the path of continuous improvement and build an efficient engineering team that is flexible in responding to business needs.

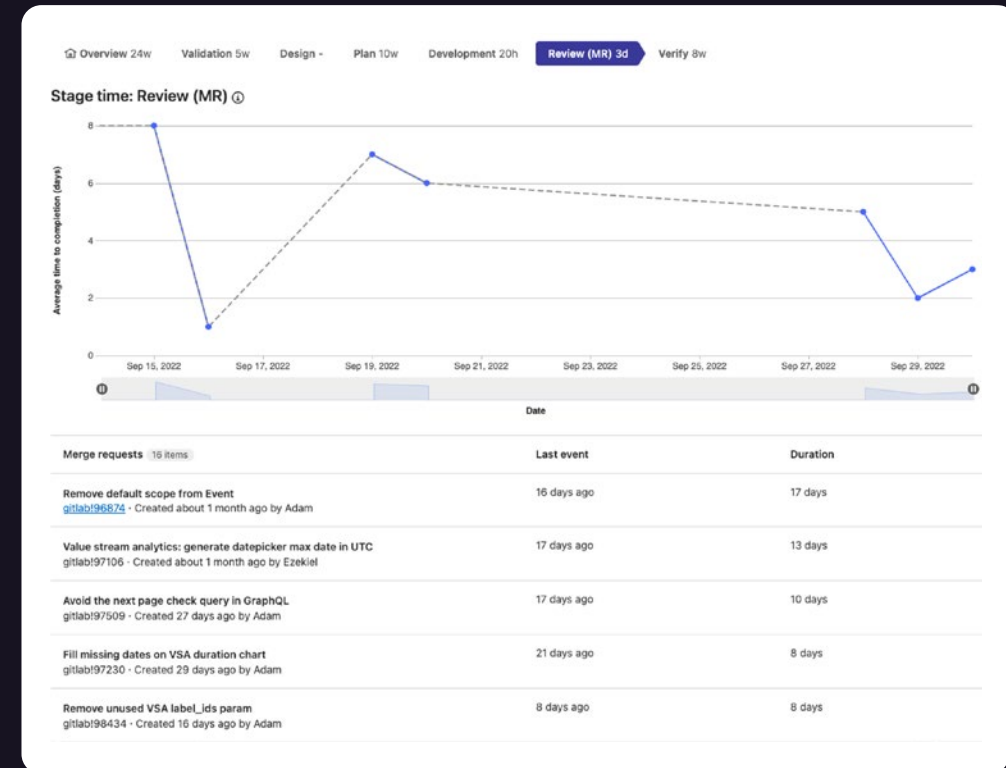


A single platform for business value insights

Driving business value isn't always a simple matter, but knowing what factors are having an impact (and how to measure them) can make it easier. Adopting a single DevOps platform can streamline this even more by:

- Providing a single place for teams to **continuously improve** code and processes
- Incorporating automated testing and comprehensive version control to facilitate code review and ensure **high-quality output**
- Consolidating complex toolchains and eliminating legacy systems to **cut costs**
- Bringing it all together with **end-to-end visibility into the entire software delivery workflow**, all in a single application

DevOps is about people. Tools power DevOps, but they aren't enough to unlock all of its benefits. Trust is what binds people together and makes effective communication and relationships possible. Without a **high-trust culture** built on transparency and information sharing, organizations are unlikely to drive higher-value software delivery and make users happy. In a typical high-trust culture, information is readily accessible, flows to the right places within the organization, and is easily findable by those who need it. With **GitLab's DevSecOps Platform**, planning, code testing, and collaboration all happen in the same system, optimizing the information flow between teams and building trust — both within the organization and with customers.



By replacing DIY toolchains with GitLab, teams can iterate faster, maintain high-quality output, reduce costs, and stay focused on activities that are driving value for customers.

Ready to start measuring the value of software delivery for your organization? **Spin up a free 30-day trial of GitLab Ultimate.**



About GitLab

GitLab is the most comprehensive, AI-powered DevSecOps Platform for software innovation. GitLab provides one interface, one data store, one permissions model, one value stream, one set of reports, one spot to secure your code, one location to deploy to any cloud, and one place for everyone to contribute. The platform is the only true cloud-agnostic end-to-end DevSecOps platform that brings together all DevSecOps capabilities in one place.

With GitLab, organizations can create, deliver, and manage code quickly and continuously to translate business vision into reality. GitLab empowers customers and users to innovate faster, scale more easily, and serve and retain customers more effectively. Built on open source, GitLab works alongside its growing community, which is composed of thousands of developers and millions of users, to continuously deliver new innovations.



