# Measuring Feature Management ROI: 7 Ways to Track Success
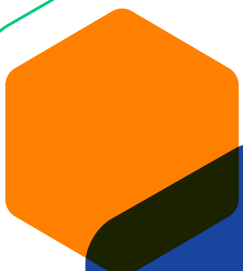
# Contents

# Introduction

DevOps is now a must-have for organizations looking to stay ahead of the curve. Yet, although 80% of developers say their company is making the move to DevOps, only 10% say they're routinely deploying code to production and just 14% achieve excellence in software development, according to recent survey data from the CD Foundation.

Feature management solutions help streamline this process by allowing teams to change application behavior on demand. But, what does this mean for return on investment (ROI)? How do businesses measure the impact of these feature frameworks to ensure they're living up to expectations? Read on to learn the basics of these solutions, dive into their key benefits, and discover seven ways to measure success.

# What is Feature Management?

Feature management solutions are designed to help DevOps teams improve development pipelines and bring more value to customers. So, what does this look like in practice?

It starts with the understanding that software releases aren't monolithic. Instead, they're made up of multiple components that work in unison to deliver the desired output. These parts vary in size; in some cases, teams need to modify small pieces of code that may be causing interface or operational issues. In others, larger features may be either the symptom or root cause of production problems.

The challenge is managing these features at scale. To accomplish this goal, developers use feature flags. These flags act as switches that allow teams to change application behavior in runtime, rather than replying. In other words, flags let your team turn features on or off without shutting down the entire process and starting from scratch.

Think of it like wiring a string of lights in parallel rather than in series—disconnect a light in series and the whole string goes dark. Do the same in parallel and only the light you removed no longer works. Traditional software development has operated in series; shutting down one feature required "rewiring" in the form of deployment to turn on or off any single feature. Feature flags, meanwhile, make it possible to build applications in parallel. If there's a potential feature problem or conflict, behavior can be dynamically altered in runtime without causing the entire application to crash.
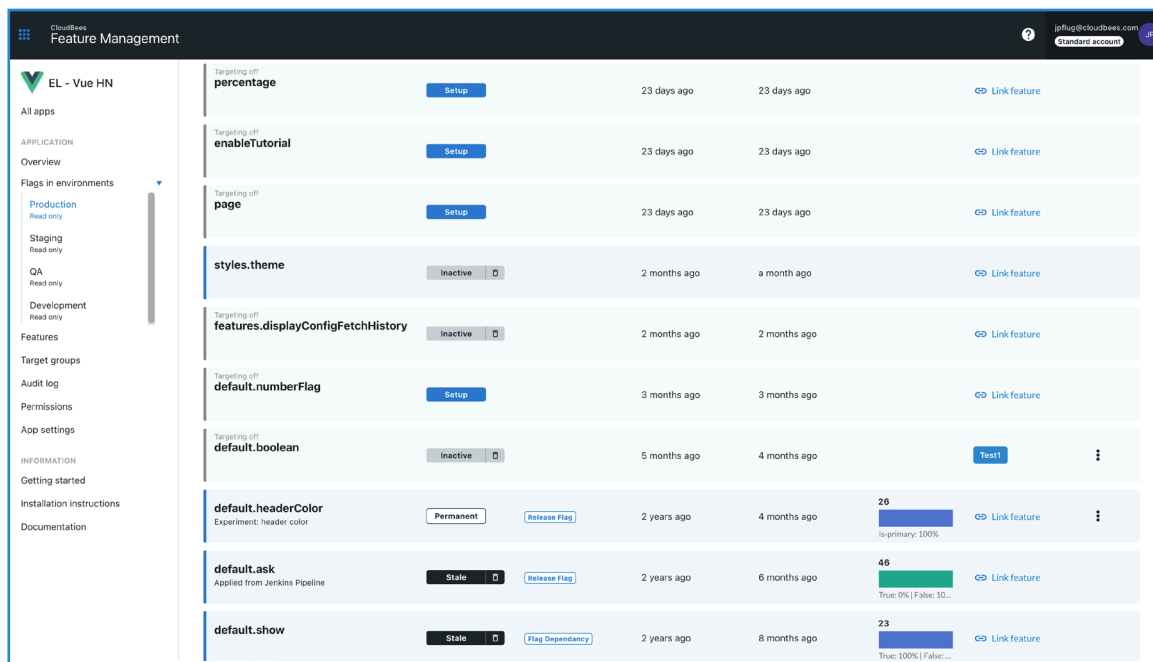
# Enterprise Benefits of Feature Flags

Although feature flags make it possible to adjust application behavior in situ, they require careful and continuous management to be effective at scale. Consider an application with multiple features that DevOps teams are evaluating simultaneously. Tracking the use of feature flags is critical for developers to ensure they understand the impact each flag has on the application itself—and how these flags interact with each other.

Feature flag solutions make it possible to manage multiple flags simultaneously, which, in turn, provides increased visibility and reliability. For businesses just making the move to DevOps, homegrown solutions tied to legacy deployments may be enough to manage feature volumes and value, but as application volumes and complexity grow, it's worth considering commercial enterprise offerings that offer both advanced and scalable functionality.

Key benefits of these solutions can allow your teams to move faster, recover sooner, boost developer productivity, and drive more innovation.

### Improved delivery times

Because management tools make it possible to easily turn features on and off—and track these feature flags at scale—businesses are better positioned to deploy new features faster and with less risk, ultimately improving the entire software development lifecycle. This reduction in time is one of the primary DevOps pillars, but it's often a challenge to achieve as applications and services become more complex.

### Quicker recoveries

Despite best efforts, no code is error-free. Feature flag management enables DevOps teams to turn off broken features during development or even when services are live. This reduces the overall impact on end users, the time required to make changes, and the complexity of restoring service if a problem occurs. Companies using these tools report significantly fewer software rollbacks, faster mean time to restore service, and faster bug fixes.

### Increased productivity

Although the concept of DevOps focuses on improved production and increased productivity, teams often struggle to reach this goal because they're spending too much time worrying about infrastructure, handling release management, and addressing rollback issues. Feature management tools can help cut down on the need for administrative and manual tasks by providing developers with on-demand data about current flag activation and its impact on apps. This gives them more time to focus on what matters: building better software.

### Enhanced insights

Ongoing innovation is a hallmark of DevOps deployments, but this isn't possible without continuous feedback from both teams and end users. Management tools make it possible to gain highly targeted feedback about how real customers use new application features—and where these features may fall short. For example, companies can use these solutions to conduct robust A/B testing or run beta programs that help pinpoint what customers like about current apps and what they want to see in future releases.

### Developer-friendly workflows

The ideal feature flag tool comes with developer-friendly workflows, such as bidirectional configuration as code with GitHub, making it easy for developers to edit flags in the environments of their choice—without the need for a dashboard.

# 7 Ways to Measure Feature Management Success

It's one thing to deploy feature management solutions; it's another to measure their impact. Although the benefits listed above describe the potential of these feature frameworks, the C-suite and team leaders need ways to gauge both qualitative and quantitative outcomes on ROI.

**Here are seven ways to measure feature flag management success:**

### 1. Monitor feature deployment

How quickly are new features being implemented and deployed? For one U.S. logistics firm, the use of feature flag management solutions allowed them to deploy 2.5 times more features into production per month, effectively cutting at least a month from existing development cycles. Increased speed of deployment is directly tied to ROI seeing as more feature deployments in a shorter amount of time means an increased ability to generate revenue.

### 2. Collect developer feedback

It's also critical to talk with developers and get their feedback about new feature flag management tools. Here's why: If companies deploy cutting-edge tools that don't effectively integrate with existing environments, developers will simply create workarounds and investments won't live up to their potential. By talking with developers before, during, and after solution deployment, businesses can ensure new tools are working as intended.

### 3. Measure recovery time

Reduced time to recovery means less time spent fixing problems and more time spent driving innovation and generating revenue. Consider data from a U.S. transportation company, which saw a fifteen-fold reduction in the time required for bug fixes—from 30 minutes down to 2 minutes or less.

### 4. Connect with end users

Along with developer impressions, it's also worth connecting with end users. Customers and clients don't have specific knowledge about new management tools, but they can communicate key information about how applications are performing, where they're encountering errors, and where improvements must be made. Creating a communications pipeline can help track the impact of feature flag tools. Has the volume and frequency of customer concerns decreased since new tools were implemented? In what areas? And where are improvements still necessary?

### 5. Assess developer activity

How much time are developers spending on new applications and features, and how much time is dedicated to fixing current problems? Management can help reduce non-development activities anywhere from 50% to 90%, depending on current DevOps environments and priorities. Tracking this metric is a good way to see if current frameworks are sufficient or if your organization needs more robust feature flag solutions.

### 6. Consider what you don't see

ROI links spending and outcome to evaluate investment value, but it's worth considering indirect benefits delivered by solutions, as well. These include reduced complexity and reliance on legacy processes, which frees up more time for developers to design, test, and deploy new applications.

### 7. Track testing benchmarks

Thorough testing is a critical part of DevOps. Framework management tools can help streamline the process without sacrificing test quality. In practice, this means cutting beta testing and A/B testing times by up to 50% to help speed time-to-market.

# Getting the Most from Your Feature Flag Management Solution

Effective feature flag management is one trend any forward-thinking software enterprise should heed to stay ahead. It makes sense: From improved delivery times to reduced risk to enhanced developer insight, these tools are essential to streamline CI/CD pipelines.

Still, getting the most from feature frameworks requires more than implementation and deployment. Businesses must evaluate the ROI of these solutions—not just on their ability to qualitatively increase performance and reduce errors, but in relation to their quantitative impact on feature deployment times, developer productivity, and testing timelines.

## Learn More

**CloudBees**®

The ROI of Using Feature Flags in Software Delivery
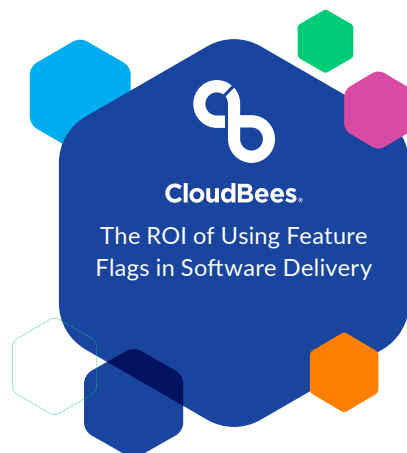
**Watch the Webinar**
*Feature Flags Across CI/CD; Expert Discussion*

**Read the Guide**
*The Ultimate Feature Flag Guide*

**Review the eBook**
*Feature Flags: Should I Build or Buy?*

## Get the Industry's best value free Feature Management Community Edition
www.cloudbees.com/products/feature-management/pricing

0522v00