The Definitive Guide to Modern Software Delivery

CloudBees.

Contents

A Profile of an Enterprise

Over the past decade, it has become clear that modern enterprises succeed or fail on their ability to develop, deliver and leverage software. Whether one is analyzing market leaders, or traditional financial services organizations alike, a close inspection reveals that adopting modern software development and delivery capabilities has enabled these organizations to increase productivity, achieve efficiency and drive innovation.

Accepting that software paves the path to a modern enterprise and successfully embracing that reality prove to be very different challenges. Many larger organizations seek to implement the modern approaches to software which are celebrated by the once smaller digital native companies. These practices include agile (interactive, user-centric planning and execution), continuous deployment (deploying every change to production), two-pizza-sized teams (small teams focused on specific features and functionality) and mantras like “fail fast and fix things.” Enterprises, however, face unique challenges in implementing these practices in a manner which sustainably drives the successful outcomes they seek.

Enterprises have the challenges of scale and complexity to overcome. Such organizations have multiple lines of business with multiple teams in each, leveraging multiple technologies coordinated across multiple functional groups, and often needing to align to provide a coherent user experience and drive shared objectives. A bank, for instance, may have a consumer banking unit, an investment banking unit, batch processing jobs, mobile applications, a web-based customer interface and so on. Each of these business units may have multiple teams who are using technology specifically suited for what they build. Some of those technologies may have been adopted decades ago, yet others have only recently been introduced. Ultimately all of their efforts has to be coordinated across different functional groups such as project management, application development, quality assurance and operations.

Scale and complexity result in a diversity of teams, tools and technologies. However, the root problem is not the diversity of the organization but rather the pervasive disconnection across teams. Disconnections between development and operations, leadership and practitioners, and front-end and back-end teams all result in inefficiency, lower productivity, failure to innovate and ultimately poor customer satisfaction.

Solving the Disconnection Problem

The reality of these pervasive disconnections is that even enterprises that adopt DevOps will still find it difficult- if not impossible - to make software delivery rapid, repeatable and scalable. The solution lies in connecting all phases of the software development and delivery lifecycle to achieve a shared and seamless process within a unified view. Ensuring that each team understands how other teams contribute to the pipeline enables the entire organization to work together to remove obstacles and achieve a common goal. By further connecting to the company's overall objectives, software teams can align with the business to drive valuable results. The best way to connect these historically disparate views is to focus on the five pillars of Software Delivery Automation: continuous integration and delivery, release orchestration, feature management and value-stream management.

Continuous Integration and Continuous Delivery (CI/CD)

Modern development teams have embraced CI/CD as the best approach when implementing automation across the software development lifecycle. The first pillar, Continuous integration (CI), means committing verified changes to code and more frequently integrating them with the application as a whole, thereby limiting the risk of conflicts and ensuring the code is working properly by itself and as part of the application. The second pillar, continuous delivery (CD), automatically and continuously verifies that code has been vetted and approved for release, ensuring the latest features are always production-ready.

The fear of deploying a breaking change into production causes many growing companies to create unnecessary processes, strict approval gates and even deployment committees or boards in an effort to prevent negative impact to customers. In the enterprise, deployments tend to be more complex and they must often adhere to rigid compliance, governance and regulatory audit needs. Building a CI/CD strategy that connects engineering teams to the stakeholders requesting and approving their changes improves trust and increases the velocity at which new and exciting features can make it out to customers.

Release Orchestration

Advanced CI/CD pipelines can be very intricate, especially if they include many different test and release stages. Thus, modern software delivery practices in an enterprise requires more than a solid CI/CD strategy. Release orchestration glues CI and CD together and brings empathy from a business point of view to all the individual pieces of the process. Done correctly, release orchestration helps IT leaders manage release pipelines and dependencies across all enterprise teams, DevOps tools and environments, enabling Ops to safely and predictably release new applications and adapt to change at any speed demanded by the business.

Where CI/CD takes software applications from code to production-ready, release orchestration is focused on putting them into production and ensuring they deliver on the business value as intended. Release orchestration enables a high level of control and visibility into complex enterprise pipelines, making it vital for modern software delivery.

Feature Management

For IT leaders, managing which features are consumable in production is a critical aspect of software releases. Feature flags, the fourth pillar of modern software delivery, give development teams more granular control over when and how features are delivered to customers, allowing teams to progressively deliver fully-proven features when the market is ready.

Traditional software release practices require engineering and business teams to work lock-step with each other in order to align business and technical requirements before a release. Unfortunately, these teams rarely work in perfect sync due to different priorities, practices and timelines. Feature flags allow features to be deployed to production when the development team is ready and revealed to customers when the business team is ready, removing the need for lock-step work and allowing each team to work at their own pace. This shift gets developers back to doing what they do best - creating the next big feature.

Value Stream Management (VSM)

Success in today's economy requires IT and engineering teams to fully align with the overarching business goals of the company. Yet all too often they're left out of the decision-making process and forced to react to, rather than anticipate, the rapidly changing landscape of customer needs and new or improved technologies. Value stream management allows engineering teams to define and prioritize their goals as concrete business results and demonstrate their value back to the business.

Closely monitoring the software delivery process allows engineers to prioritize work in alignment with business objectives. Forcing software teams to react to executive mandates often leads to shadow engineering, build-up of technical debt and burnout. To avoid these pitfalls, all teams should be positioned to understand exactly what business objective their work drives and how the tasks they perform map back to value added to the company.

Best Practices to Support the Five Pillars

In order to break down team silos and connect both people and processes, it is imperative that all five pillars of modern software delivery be implemented. Deceptively attractive as standalone practices, implementing one or more separate of the whole simply shifts the problems out of sight and into other areas of the software pipeline. Only by understanding the entire lifecycle of a desired change from both technical and business perspectives can organizations achieve the speed, safety and value offered by modern software delivery.

Integrated, Automated, End-to-End Software Delivery

By using automation and orchestration to connect across diverse teams, technologies and function groups, enterprises can achieve end-to-end visibility into the software delivery lifecycle. This environment ensures repeatable, reliable and rapid hand-offs which reduce the risks and cost of software delivery. Embracing the complexity of the enterprise software landscape by implementing shared platforms, with a shared process and a shared view, instead of redesigning the whole software delivery toolchain, enables teams to use the tools and technologies they excel with and the organization to leverage existing investments.

Software Delivery Automation and the Future of Modern Software Delivery

How do enterprises make this a reality? The answer is Software Delivery Automation, connecting the end-to-end software delivery lifecycle across the organization's teams tools and technologies with a unified automation and orchestration layer. Specific teams purpose-fit tools and technology into an end-to-end delivery platform.

CloudBees Software Delivery Automation solutions enable enterprises to leverage connected, proven solutions to overcome the disconnection problem. Supporting the five pillars of modern software development with continuous integration, continuous integration and delivery, release orchestration, feature management and value stream management solutions. By leveraging the CloudBees platform, these software delivery automation solutions reduce the risk and cost of software delivery and allows organizations to get the most out of existing investments.

Software Delivery Automation enables organizations to:

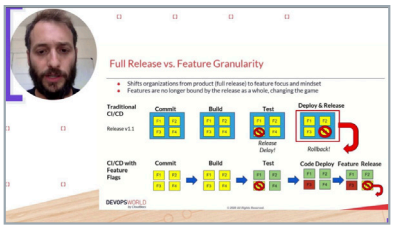
- Innovate, mitigate risk and shift faster
- Ensure security and compliance
- Optimize your software development and delivery lifecycle, end-to-end

So that they can:

- Deliver on time and on budget, consistently at scale
- Build a culture of freedom and innovation
- Modernize software delivery for an entire diverse portfolio

By providing deep integration between market-leading solutions for CI, CD, release orchestration and feature flag management across all of your teams and tools, CloudBees Software Delivery Automation solutions enable any organization to implement all five pillars of modern software delivery at scale, and across a complex diverse enterprise portfolio.

Learn More



Replay the DevOps World Session
The Future of Feature Flags and CI/CD: Shifting to Feature-First Organizations

Read the Whitepaper
Grow Continuous Delivery Maturity Using Feature Flags

Watch the Webinar
Evolving Continuous Delivery Using Feature Flags

Learn More:

www.cloudbees.com/software-delivery-automation

CloudBees CI is built on top of Jenkins, an independent community project. Read more about Jenkins at: www.cloudbees.com/jenkins/about

0221v00

CloudBees, Inc.
4 North Second Street | Suite 1270
San Jose, CA 95113
United States
www.cloudbees.com
info@cloudbees.com