Optimize DevOps with Application Release Orchestration

CloudBees®

Digital transformation is ubiquitous. Getting quality software into production quickly and efficiently is a major priority for organizations of all types. Yet many find that development teams' agile pace of software delivery and new technology adoption has outstripped the ability of operations teams to keep up. This cadence and skills mismatch puts a speed limit on innovation, puts transformation success at risk, and leaves release managers in a constant state of release anxiety.

To keep up with this pace of innovation, organizations need to:

- Let development teams freely experiment and adopt new technologies.
- Help operations teams safely absorb that new technology and easily move it into production when the business demands.
- Provide infrastructure, operations and release managers an easy way to ensure quality, integrity and security throughout the process, no matter the scale.

This eBook describes how DevOps practices, and application release orchestration (ARO), help make releases systematic, predictable and repeatable to eliminate release anxiety and allow organizations to release on business demand.

DORA State of DevOps Report, 2019:

"The proportion of high performers has grown year over year, showing that the industry is continuing to improve. Comparing the elite group against the low performers, we find that elite performers have:"

- **208 times more** frequent code deployments.
- **106 times faster** lead time from commit to deploy.
- **7 times lower** change failure rate (changes are 1/7 as likely to fail).
- **2,604 times faster** time to recover from incidents.

What is DevOps?

DevOps is the methodology that fosters collaboration between the teams that create and test applications (development) with those that maintain them in production environments (operations). It aims to help an organization rapidly produce software products and services while maintaining governance.

What is Application Release Orchestration?

Application release orchestration (ARO) is a new approach to release automation that puts an emphasis on organizational and toolchain agility, ensuring teams can easily adapt to and take advantage of new technologies and situations safely.

CloudBees CD is a powerful application release orchestration platform that is designed to integrate easily with current and future toolchains, while allowing teams to model their environments, pipelines, applications and releases. This model-based approach allows teams of any size to orchestrate complex releases, perform deployment automation and streamline pipeline and environment management.

Why ARO matters

Application release orchestration matters because getting software to market faster is the new secret weapon. It helps IT operations release on business demand by enabling them to:

- Model repeatable, predictable and manageable workflows that provide a rehearsable, well-worn path to production so others can follow.
- Accelerate time to revenue with the ability to show results quickly through the creation, versioning and sharing of multiple delivery pipelines and environments.
- Reduce and control costs through customizable automation that eliminates steps, especially errors caused by manual handoffs and scripts.
- Easily adapt to shifting market demands, technology innovation and process change.
- Confidently speed up cycle times by delivering predictably at any speed and scale.
- Improve management controls and governance through structured, cross-functional automation with role-based access controls, integrated planning and activity tracking.
- Shift quality and security left by detecting – and correcting – problems before they get into production.

5 Pillars of Application Release Orchestration

Many of the largest and most complex software organizations leverage application release orchestration solutions from CloudBees to automate and streamline their software deployment process.

The five pillars of ARO that ensure DevOps success are:

1. Model and automate everything
2. Provide environments and automation as a service
3. Adopt new technologies safely
4. Monitor and track releases
5. Build in security and compliance

1 Employ Application, Pipeline and Environment Models

Instead of relying on vast collections of brittle, hard-coded scripts and manual processes (the two most common sources of failed deployments), base your DevOps success on models. Application, pipeline and environment models abstract the who, what and how of application delivery to promote the reuse of well-rehearsed work across toolchains or environments without requiring manual commands or deployment time script changes. Team members no longer need to be subject matter experts in everything if they want to try something new, like deploy to a Kubernetes container. Employing models also ensures that the same process is used consistently in every environment and stage, and roles remain separate.

Benefits of Models

No need to re-invent the wheel because models are repeatable, auditable and manageable

Eliminates unplanned work and heroic efforts because standard practices are used every time

Lays the groundwork for flexibility because models are easy to version, test, refactor or extend

Appropriate approvals and separation of duties become the norm because best practices are standardized for each and every release

2 Provide Environments and Automation as a Service

Digital transformation at scale requires team members to test existing applications and onboard new applications and teams quickly, safely and consistently. With a self-service catalog of standard environments and components, team members can eliminate the wait and deploy any number of applications or systems to on-premise or cloud environments with one click.

Based on models, these are easily versioned and refactored, so updating them to accommodate new technology or process is no longer dangerous. Faster deployment means faster feedback, so team members will respond to errors and changes gracefully and more quickly.

Benefits of Environments and Automation as a Service

Rapidly onboard new teams, pipelines and applications because domain-specific knowledge is not a requirement

Predictability is the norm because standardization ensures consistency and captures drift at each stage in the process

Experimentation and incremental enhancement is easy because team members start with what they know and make changes as they gather more data

Governance is a breeze because automated processes and immutable components easily meet compliance and regulatory needs

3 Evaluate and Adopt the "New"

The tools of today are not the tools of tomorrow. Being able to respond to new process requirements (like GDPR), new technology (like Kubernetes) and new architectures (like serverless) can be a differentiator for many organizations. Fortunately, a non-prescriptive model-based approach makes this type of experimentation and change safe. Models encourage development teams to try out new tools and techniques, and rapidly iterate on the company's "secret sauce." This helps teams safely introduce and test new tools without causing disruptions or bottlenecks for downstream teams or environments.

Benefits of Adopting New Tools and Methods

Future-proof organizations because adapting to change is non-disruptive

Take the rocket science out of new technology and give operations teams the confidence to deploy new technology into production without anxiety

Confidently embrace and roll out new use cases or processes because it is encapsulated in a rehearsable model that has been tested and proven to work

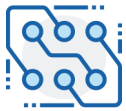
4 Monitor and Track Releases

It's 12:30AM. Do you know where your release is? Or what spreadsheet it's in?

Self-reporting, email and spreadsheets are so last century. DevOps success relies upon 360° visibility into everything: release status, success rates, bottlenecks, dependencies, environments, deployments and all associated point tools. Build centralized and shared dashboards to track progress and identify bottlenecks, errors and waiting time.

Give teams guaranteed, shared access to resources like environment reservations, calendaring and blackout/ maintenance windows. Provide teams detailed analytics, with historical data across multiple releases, so they can identify trends and hotspots.

Benefits of Monitoring and Tracking Releases



Faster and better decisionmaking because there's complete visibility throughout releases, pipelines and third-party tools

Inter-team communication improves because teams share the same language and status of releases

Success rates go up because problems and bottlenecks can be anticipated and mitigated

Managing complex distributed deployments, like microservices, is easier because key metrics are readily available

5 Build In Security and Compliance

Security and quality is everyone's responsibility. An ARO platform makes security the path of least resistance with reusable security and compliance pipelines designed to work across teams, applications and environments. Shift security left by integrating new security tools and compliance policies into the pipeline. Limit risk exposure with an inventory of all environments and deployed components. Create versioned, controlled-access pipelines that ensure process integrity and auditability. Test the pipeline for efficacy by injecting failure, like unapproved components, at any stage. Constantly improve the pipeline by incrementally adding additional security tools or policies in a managed way.

Benefits of Built-in Security and Compliance

Security and compliance are no longer bottlenecks or an after-thought, because they are integrated and automated

Governance and reporting is no longer a chore because policies are automatically enforced and changes to the pipeline are automatically recorded

Problems are detected – and corrected – earlier because tests and gates are executed at the right time, in the right sequence

Automatic anomaly/ drift detection ensures compliance and identifies vulnerabilities and bypassed processes

Incident response time and security patching across hybrid environments is accelerated, with 360° view of exposure radius and release progress

Scale doesn't limit the business because it is baked-in and tested

Summary

Application release orchestration (ARO) from CloudBees eliminates release anxiety by making the process of delivering software consistent, repeatable and auditable. It future-proofs organizations by making it easy to embrace change, onboard new teams and applications and experiment with what's new while staying secure. It improves decision-making with comprehensive insight and control across the entire lifecycle. ARO connects, streamlines and speeds the flow of development, testing and deployment to meet the needs of the business and comply with regulatory constraints.

About

CloudBees is the industry's leading DevOps technology platform delivering the world's first end-to-end continuous software delivery management system. CloudBees enables developers to focus on what they do best: Build stuff that matters—while providing peace of mind to management with powerful risk mitigation, compliance and governance tools.

Used by 50% of the Fortune 500, CloudBees is helping thousands of companies harness the power of continuous everything and gets them on the fastest path from great idea, to great software, to amazing customer experiences, to being a business that changes lives.

Visit CloudBees at www.cloudbees.com.

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

0820v02

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