

The CI/CD Buyer's Guide

By: Roxanne Williams . Content Strategy Manager

© Harness Inc. 2021

Table of Contents

04 What is CI/CD, and Why Should I Care?

05 Why is it Important to Invest in the Right CI/CD Platform?

06 Continuous Integration Tools

07 AWS CodeBuild

08 CircleCI

09 Concourse CI

10 Drone

11 Travis CI

12 Continuous Deployment Tools

13 Argo CD

14 Armory

15 Chef

16 Digital.ai

17 GoCD

18 IBM Cloud DevOps

19 Micro Focus

20 Octopus Deploy

21 Spinnaker

22 CI/CD Tools

23 Azure DevOps

24 Bamboo

25 Bitbucket Cloud

26 Buildkite

27 CloudBees

28 Codefresh

29 GitLab

30 Google Cloud Build

31 Harness

32 Jenkins

33 Jenkins X

© Harness Inc. 2021

Opening Words

Picking a CI/CD platform is, without a doubt, one of the most daunting things anyone will do in their career as a technologist. There's so much to think about: what features do you need right now? What features will be needed in the future? Is pricing an obstacle? Which vendor is best-suited and provides the best support? Add to that maybe a hundred more questions and you'll have a good idea how hard this decision can be.

Of course, choosing the right technology is critical to overall success—but it's not all about technology. We realized a long time ago that successful CI/CD requires both technology and cultural shifts. When migrating from homegrown (or brand new) solutions to a vendor-provided software delivery platform, it can be really hard to let it go and embrace different technology. DevOps/engineering teams often pour their heart and soul into solutions they've extended/created from scratch, and it's a hard pill to swallow to see a CI/CD vendor come in and 'destroy' that—even if it's to make things easier and better for everyone.

CI/CD is just as much about adoption and culture as it is about the tech itself. Even though this buyer's guide will focus on technology, make it a point to involve multiple teams in the evaluation process and buying decision. Have DevOps test-drive platforms, and make sure that engineers get hands-on time building and running pipelines. Get your teams on board with the process. Adoption will be much easier if many stakeholders have a hand in picking the right tool that'll meet their needs—current and future.

Here's one last piece of advice: be sure to evaluate every vendor or project on their ability and likelihood to be a good partner. When selecting a vendor, it's important to create a mutually beneficial partnership. Look for a solution provider that knows how to—and is willing to—help drive successful CI/CD adoption across your organization. This will obviously be different if you select an open-source solution. In which case, we'd recommend looking at community support and seeing if these vendors do have support plans you could pay for.

The Harness Team

© Harness Inc. 2021

What is CI/CD, and Why Should I Care?

To put it succinctly, a CI/CD platform is a comprehensive set of tools that help your developers, engineers, and DevOps practitioners package and deliver software to the end users. A well-rounded CI/CD platform enables an automated software delivery process, from code check-in to post-deployment verification and beyond.

Diving into specifics, a basic CI/CD platform will need the following modules:

- **Continuous Integration**—Decrease lead time for new builds.
- **Continuous Delivery**—Always have a functional new release ready to deploy.
- **Continuous Deployment**—Deploy new release at will.

Those are the basics that will make it possible to build, package, and deploy new software releases. While the basics are a good starting point, there is more to consider when building out automated software delivery pipelines:

- How fragile are the pipelines? Are there a lot of integration points that are manually coded so that minor changes end up in complete pipeline outages?
- How long does it take to build a pipeline and onboard a new service?
- Is it easy to make a change, or is the maintenance of the pipelines more work than not having them at all?
- Who can execute a deployment to which environment?
- Is there an audit trail for regulatory or other purposes?
- What was the cloud cost of each change or new service? Is over-provisioning your cloud resources occurring and wasting profits?
- How quickly are bad deployments identified and resolved?

Taking the above considerations into account, a CI/CD platform should, in addition to the basics above, include the following:

- Automated Verification—Immediately know the health of a new release.
- Enterprise Governance—Implement granular access control and know who performs every pipeline action.

To learn more, read the full article: "What is a CI/CD Platform and Why Should I Care?" >

© Harness Inc. 2021

Why is it Important to Invest in the Right CI/CD Platform?



—
Marcus Bastian
DevOps Manager
leaflink

"CI/CD is a crucial function of software delivery. With so many choices out there, it's imperative to do research into potential vendors and ensure all the features you need for the long-term are available. Put serious thought into scalability, reliability, extensibility, governance, and security. If you subscribe to the NoOps philosophy, one of your core objectives should be for everything to be self-service. To do that successfully, things have to be fairly simple. Fewer systems = reduced complexity.

You must also decide whether to go with separate CI and CD vendors, or have it all under one roof. When making that decision, take into account secrets and configurations—some may need to be shared between two vendors, if you go that route. Having them in the same platform, with built-in secrets management, can be beneficial. It's also important to think about the possibility of a security breach—a platform like Harness can manage security for you, and you can place your delegates in your private infrastructure, preventing your most prized secrets from being compromised.

> *"Don't waste time on temporary solutions—something like Jenkins is great to get started in, but complex pipelines require tons of DevOps hours of maintenance, writing scripts, and installing plugins, leading you to over-engineer to make up for a lack of functionality. 'Free' doesn't always mean 'good.'"*

As someone who has used a plethora of CI/CD platforms, such as Jenkins, Argo CD, CircleCI, Travis CI, Drone, and Harness, I can say that the functionality, simplicity, and user experience of Harness will help you spend more time writing software instead of worrying about how to deploy it."

© Harness Inc. 2021

01
Continuous Integration Tools

Key

At Harness, we've found that most CI/CD challenges come from the below four pain points. We've built our buyer's guide with these in mind. The icons will act as a key for all reviewed vendors. Look to the right of the vendor's name to see which pain points the solution most addresses.

Velocity
Time-to-Market
Developer Velocity & Onboarding
Cloud & Microservices Migration

Quality
Downtime & Incidents
Defects & Regressions
Performance

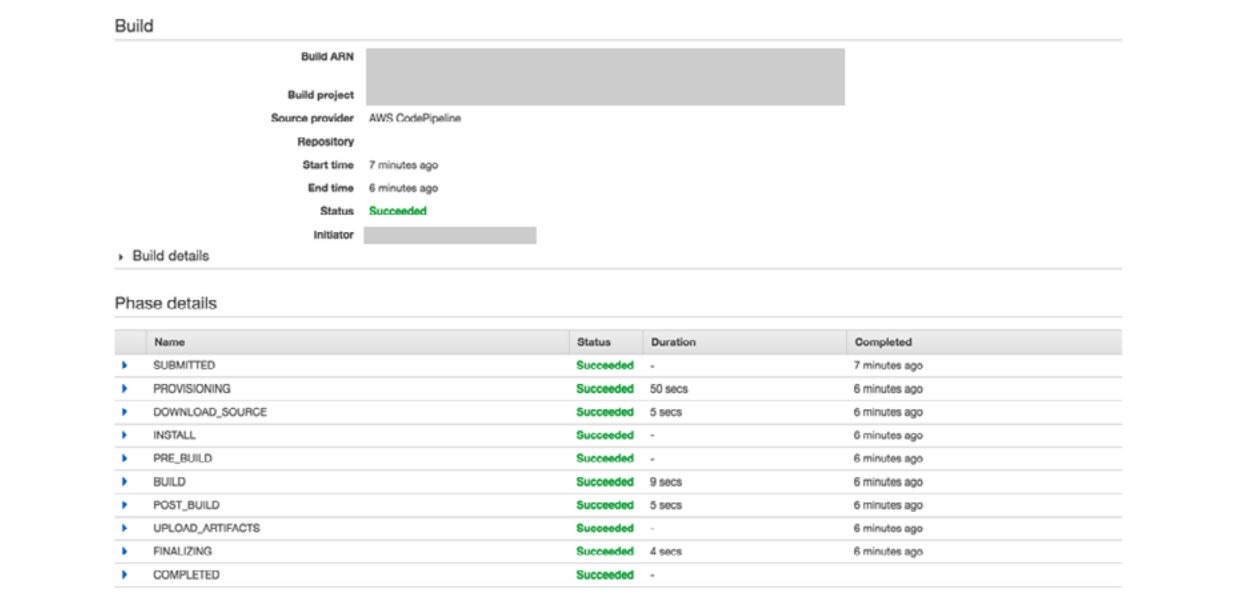
Governance
Standardized Deployments
Auditing & Compliance
Role-Based Access Control

Efficiency
Developer Productivity
Cloud Costs

© Harness Inc. 2021

AWS CodeBuild

For DevOps professionals who use AWS as their cloud provider, trying out the different offerings that are native to AWS is a common initial path. CodeBuild is a fairly robust, fully-managed CI tool that compiles source code, runs tests, and produces software packages that are ready to deploy. There's no need to manage/scale your own build servers, and builds are run concurrently. CodeBuild also has pre-made build environments, which can be useful if you're not at the point where you can build custom environments.



Pros

As stated above, CodeBuild can be a good choice for organizations trying to keep as few vendors as possible—having everything 'under one roof' can be appealing.

- Integrates with CloudWatch for monitoring and observability.
- Build agents can be configured to automatically scale to meet demands.
- Docker-based build environment, YAML-based configuration.

Cons

A common issue that developers face when leveraging CodeBuild is the complexity of the initial setup. The UI, similar to AWS' other products, doesn't lend itself to making it any easier, and a 'steep learning curve' is a recurring theme in user reviews.

- Frustrating, complex initial setup.
- Not very flexible or customizable.
- Sparse UI that has a steep learning curve.

Pricing

Per minute. Depends on a few factors, namely your region, instance size, memory, CPU, and Windows/Linux. "To the right, you will find a pricing sheet for Linux instances in the US-West (Oregon) region.

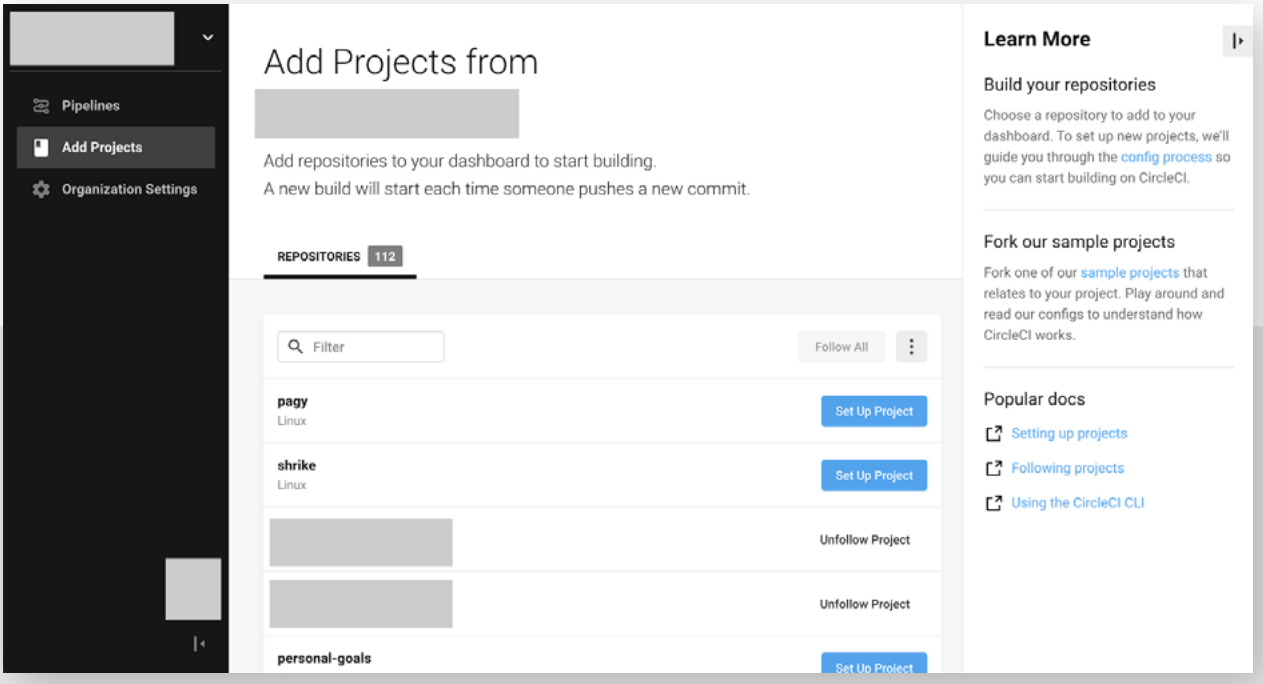
Based on this information, if you execute 100 builds in one month using a general1.large where each build runs for 5 minutes, you could expect to spend $10 for the month on AWS CodeBuild.

Compute instance type	Memory	vCPU	Linux price per build minute
general1.small	3 GB	2	$0.005
general1.medium	7 GB	4	$0.01
arm1.large	16 GiB	8	$0.015
general1.large	15 GB	8	$0.02
general1.2xlarge	144 GiB	72	$0.20
gpu1.large	244 GiB	32	$0.65

© Harness Inc. 2021

CircleCI

CircleCI was built for speed and scalability. Build configurations are stored in your application repo as YAML, making it easier to track changes and contribute to your testing process. The Orb system allows you to extend pipelines using third-party functionality with minimal effort. These features create a standardizable CI solution that can easily be adopted. CircleCI definitely rates as one of the better CI tools currently available.



Pros

Modern tool with many desirable features in a CI tool that supports many integrations, providers, and more.

- Nice UI, flexible build environments, easy to adopt.
- Supports RBAC for access to secrets.
- Supports a variety of build agents with different resource allocations.

Cons

The biggest risk in selecting CircleCI is the amount of downtime and slowness the platform experiences. Users report semi-frequent downtime and the need to stop and restart jobs that take a long time to process.

- Jobs can take a long time and can be buggy if modified.
- Establishing reusable components requires using shell scripts or writing custom Orbs.
- Limitations around Docker executor.

Pricing

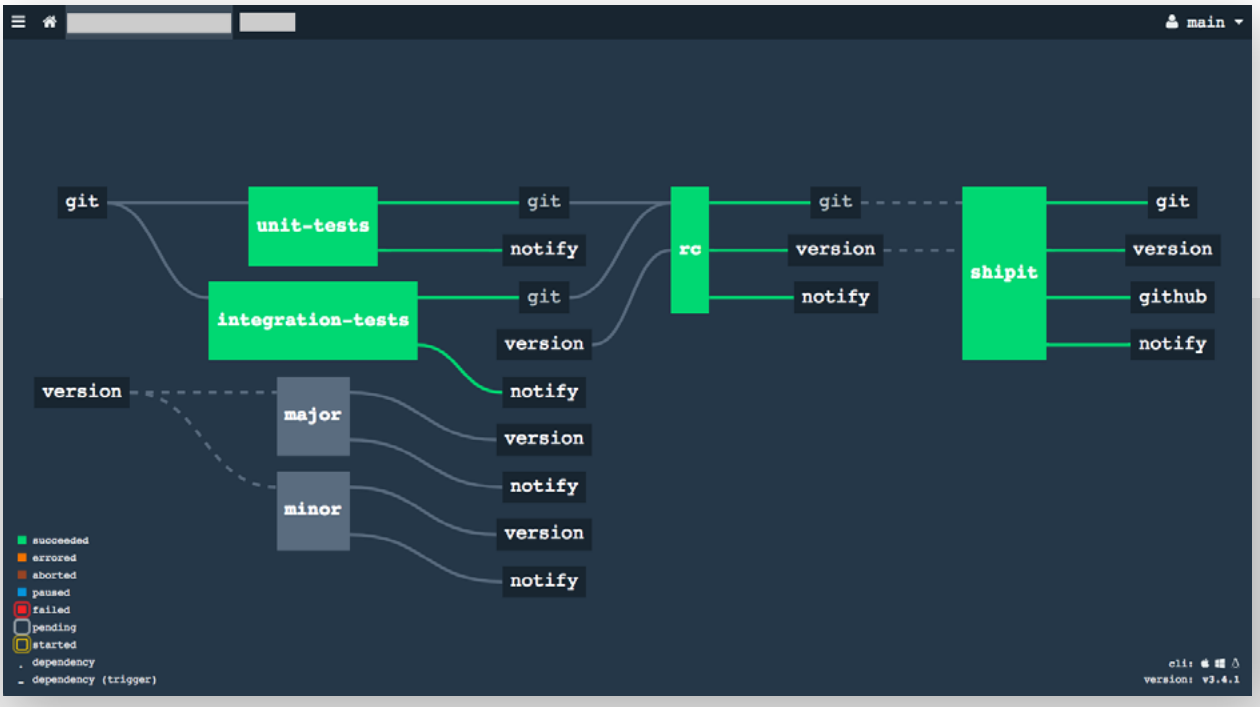
CircleCI offers a few different tiers to purchase from. If you're a smaller-scale shop with few developers, you might be able to get away with the Free or Performance plan, but the more tests you add to your application and the more developers that push code, the pricier it gets. It's sort of the same model as AWS CodeBuild—the more resources you use, the more you pay.

> The biggest risk in selecting CircleCI is the amount of downtime and slowness the platform experiences.

© Harness Inc. 2021

Concourse

Concourse was created by two engineers in 2014 after trying various CI tools that always fell short. They proceeded to open-source the project on GitHub, where it sits at 5.5k stars at the time of this writing, and is constantly being updated by the creators and contributors. There are also lots of support options through the immaculate Docs section, GitHub, Discord, and more. While it has a steeper learning curve than some, the scalability and portability of the pipelines is the entire basis of the tool—everything runs in Docker, making catastrophic failures a thing of the past.



Pros

Open-source solution with surprisingly modern UI for such a young tool. Fast, lightweight, scalable, and numerous support options.

- Open-source, gorgeous UI.
- Everything runs as a container.
- Scalable, flexible, fast, simple configuration management.

Cons

As this is a newer tool, there are features that other CI tools have that haven't been added to Concourse CI yet. However, it is actively updated and maintained so these features may come with time.

- As with anything open-source, there are constantly bugs that need to be addressed.
- Lacks basic features, requires time coming up with workarounds.
- Lots of worker stability issues, and Docker image resource doesn't work as intended (it is currently being deprecated in favor of Prototypes).

Pricing

A whopping $0. Therein lies the beauty of open-source. We would be remiss if we didn't remind you, though, that "free" doesn't always mean "free." There will always be costs associated with CI/CD, even if open-source. Maintenance comes at a cost, in terms of FTEs.

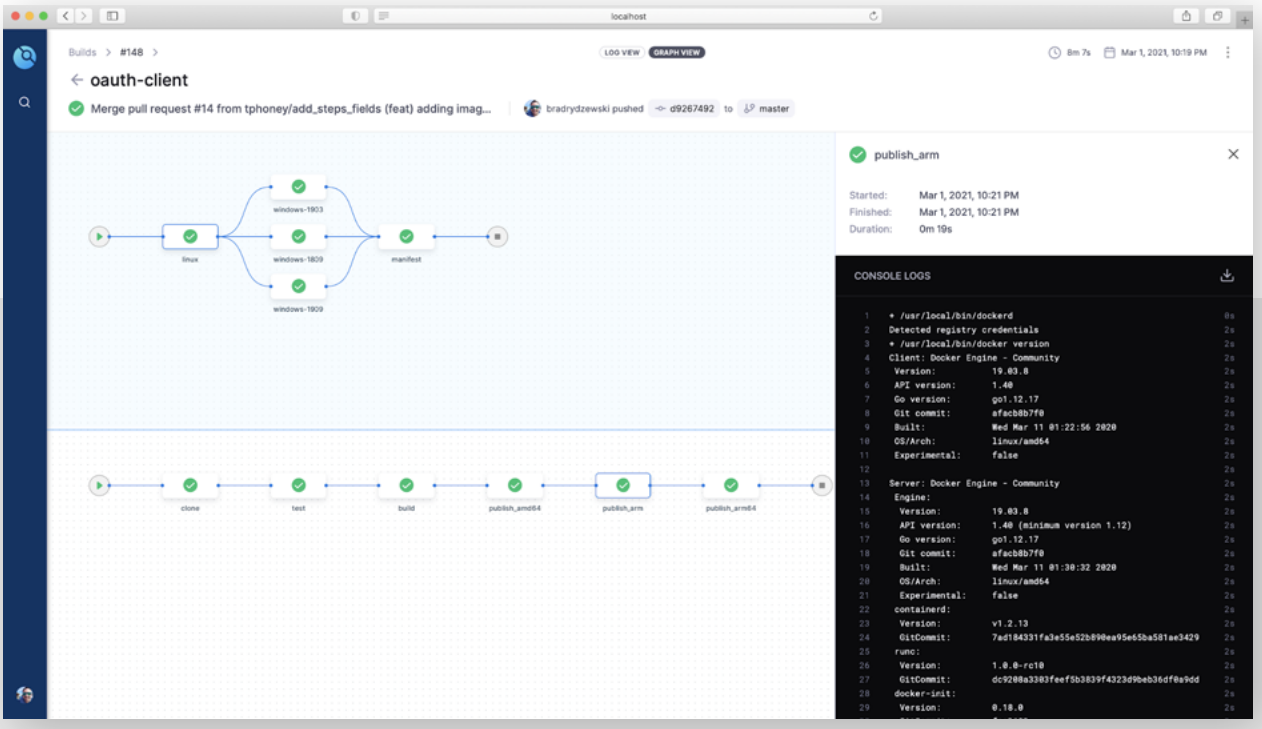
—

While it has a steeper learning curve than some, the scalability and portability of the pipelines is the entire basis of the tool—everything runs in Docker, making catastrophic failures a thing of the past.

© Harness Inc. 2021

Drone

Back in 2012, Brad Rydzewski founded an open-source project called Drone. Brad created Drone to empower engineers with a self-service CI platform that was container-native, scalable, and simple to use. Drone flourished, garnering over 100 million pulls on DockerHub, 50k active users, 275+ contributors, and a whopping 22.5k stars on GitHub. Harness acquired Drone in August 2020 to further its mission of simplifying software delivery for engineers and DevOps teams.



Pros

Drone packs a mighty punch for such a lightweight tool. It is feature-rich, with many options/integrations for each feature, allowing you to select your preferred third party vendor.

- Every step runs in its own container.
- No scripts required, every pipeline is configuration as code.
- Installs in 5 minutes, and there are zero plugin dependencies.

Cons

Governance capabilities could be expanded. Good secrets management options, but users could benefit from an RBAC feature.

- Not as many plugins as an old solution like Jenkins.
- Not part of the Harness platform, so governance capabilities are not as robust.
- No way to visually build pipelines like you can with Harness CIE—there is a visual display when the pipelines execute, but no visual builder.

Pricing

It's open-source, so there are no costs upfront—only in terms of maintenance, etc. At .25 FTEs required annually for maintenance, even those costs are low when compared to Jenkins and other more heavyweight open-source CI tools.

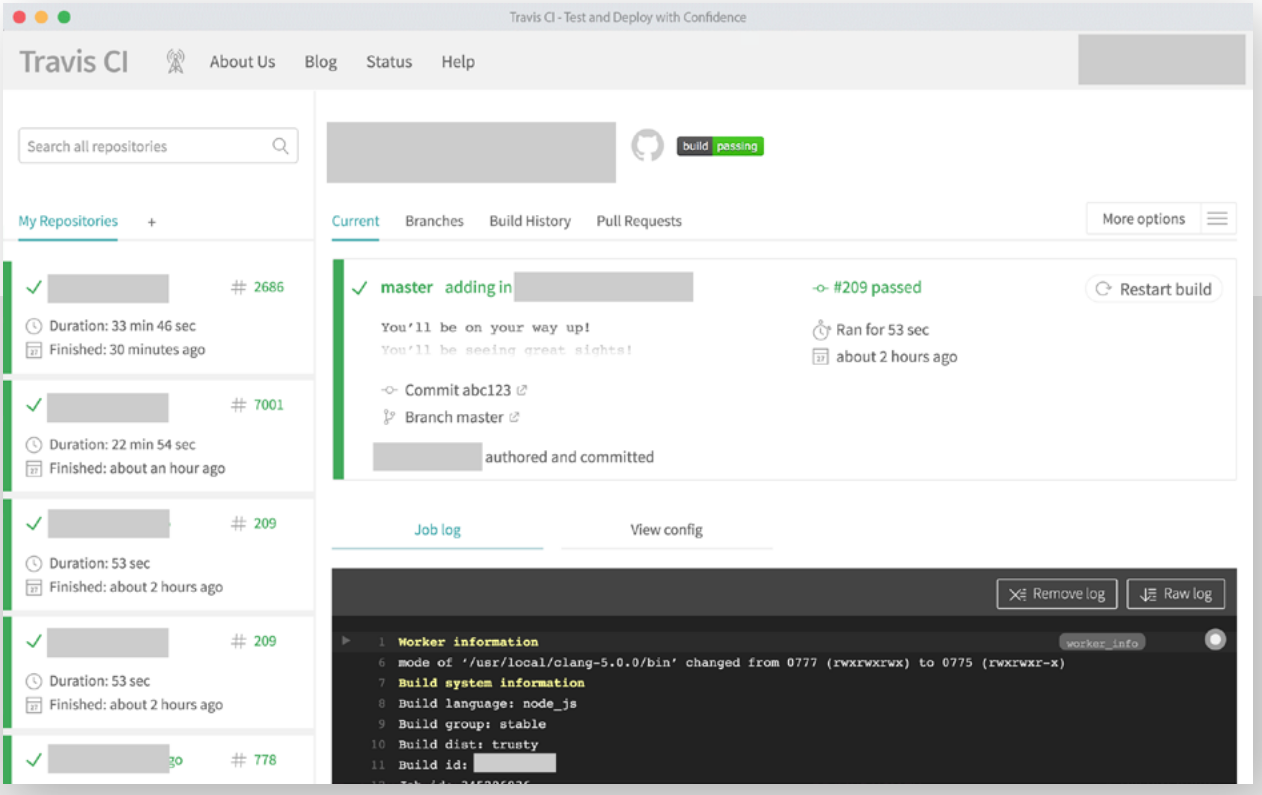
Drone packs a mighty punch for such a lightweight tool. It is feature-rich, with many options/integrations for each feature, allowing you to select your preferred third party vendor.

© Harness Inc. 2021



Travis CI

Travis CI was the first CI-as-a-service solution. It was designed for open-source projects: testing them is, and always will be, free—the creators of Travis CI offered this as a way to give back to the community. Additionally, the tool offers a streamlined experience that is easy to set up and use. Configurations are created using YAML. Travis CI lacks many of the enterprise-grade features of other CI tools (think: security and governance), but that, in turn, means it requires little effort to maintain.



Pros

Travis CI provides great language support and allows you to work in your preferred programming language. No-frills tool with easy maintenance requirements.

- Testing open-source projects is completely free.
- Initial setup takes seconds, UI is appealing, integrations are easy to get going.
- Supports over 30 languages, comes with preinstalled databases.

Cons

Travis CI is at the back of the pack when it comes to features, especially when it comes to governance. Their focus seems to be moreso on open-source projects, so while there is an enterprise-grade version, we'd still steer clear if you're a larger organization with strong security and compliance needs.

- Lacks robust governance and security—for example, adding secrets to a project is inordinately hard.
- Lacks many other platform/enterprise-grade, modern features.
- Sometimes unreliable—expect downtime.

Pricing

Travis CI bases their pricing on the amount of concurrent jobs you run. 1 concurrent job starts at $69, 2 concurrent jobs starts at $129, and 5 concurrent jobs starts at $249. They also have a free plan that could suffice for small personal projects. Lastly, there's an enterprise pricing option at $8,000 for 20 licenses, with additional licenses sold in packs of 20.

Travis CI is at the back of the pack when it comes to features, especially when it comes to governance. Their focus seems to be moreso on open-source projects, so while there is an enterprise-grade version, we'd still steer clear if you're a larger organization with strong security and compliance needs.

© Harness Inc. 2021

02
Continuous Deployment Tools

Key

At Harness, we've found that most CI/CD challenges come from the below four pain points. We've built our buyer's guide with these in mind. The icons will act as a key for all reviewed vendors. Look to the right of the vendor's name to see which pain points the solution most addresses.

Velocity
Time-to-Market
Developer Velocity & Onboarding
Cloud & Microservices Migration

Quality
Downtime & Incidents
Defects & Regressions
Performance

Governance
Standardized Deployments
Auditing & Compliance
Role-Based Access Control

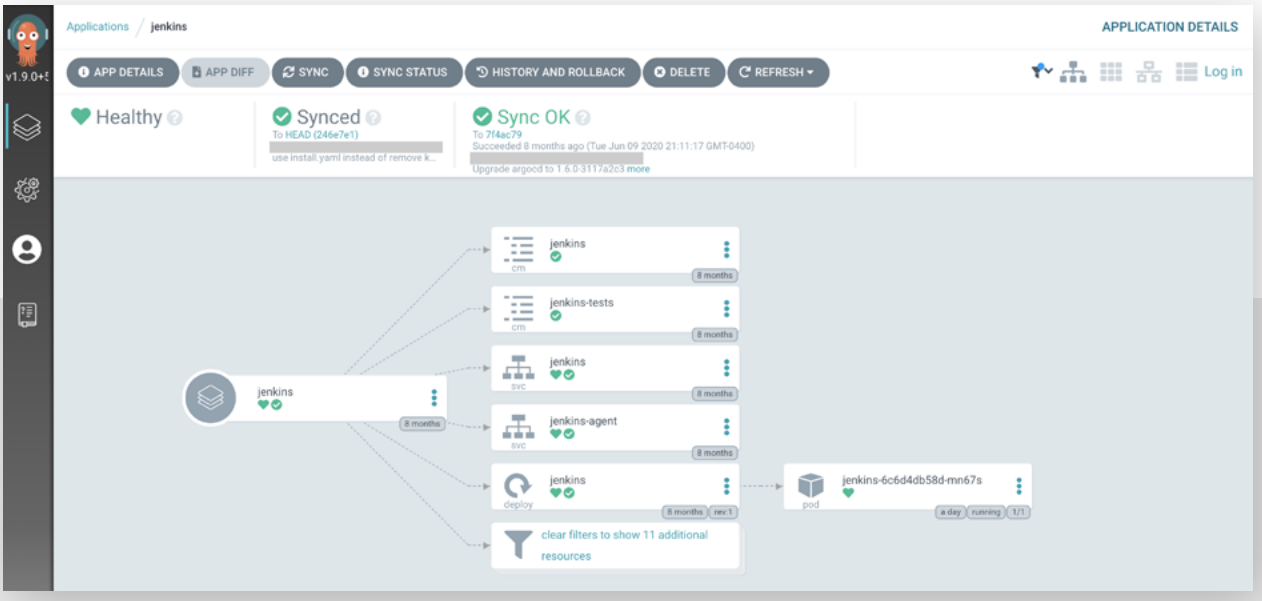
Efficiency
Developer Productivity
Cloud Costs

© Harness Inc. 2021



Argo CD

Argo CD automates Kubernetes deployments using GitOps technology. With support for a wide variety of config management tools, SSO integrations, and web hook integrations, Argo CD is a feature-rich deployment tool. It reduces administration toil by making application definitions, configurations, and environments declarative and version-controlled. Argo CD's stated goal is to make application deployments automated, auditable, and easy to understand. It boasts 5k stars on GitHub.



Pros

Built on Kubernetes, Argo CD offers a modern take on Continuous Deployment. It's a very customizable, self-service tool that provides waves so you can push more of your deployment steps into Kubernetes, eliminating the need for CD in Jenkins.

- Built-in GitOps capability, flawless Kubernetes integration.
- Supports templating with Kustomize, Helm, and more.
- Multi-cluster and multi-tenancy capabilities, RBAC, SSO, web hooks—very robust features.

Cons

Argo CD's modern features may become an impediment—the biggest one being Argo CD's auto-deploy capability.

- Might require changes in how deployments are triggered: Argo CD "watches" your Docker and Helm chart repos and it can auto-deploy when changes are detected.
- Installation required in every cluster—you will have 5 installs if you have 5 clusters.
- Doesn't have native secrets management, but can be achieved with Vault or other secrets management tools.

Pricing

Open-source! This is easily one of the CD platforms we can recommend if you're looking for an open-source solution. It's so feature-rich right out of the box that no other open-source CD tools currently on the market really compare. However, do keep in mind that open-source doesn't mean free.

—

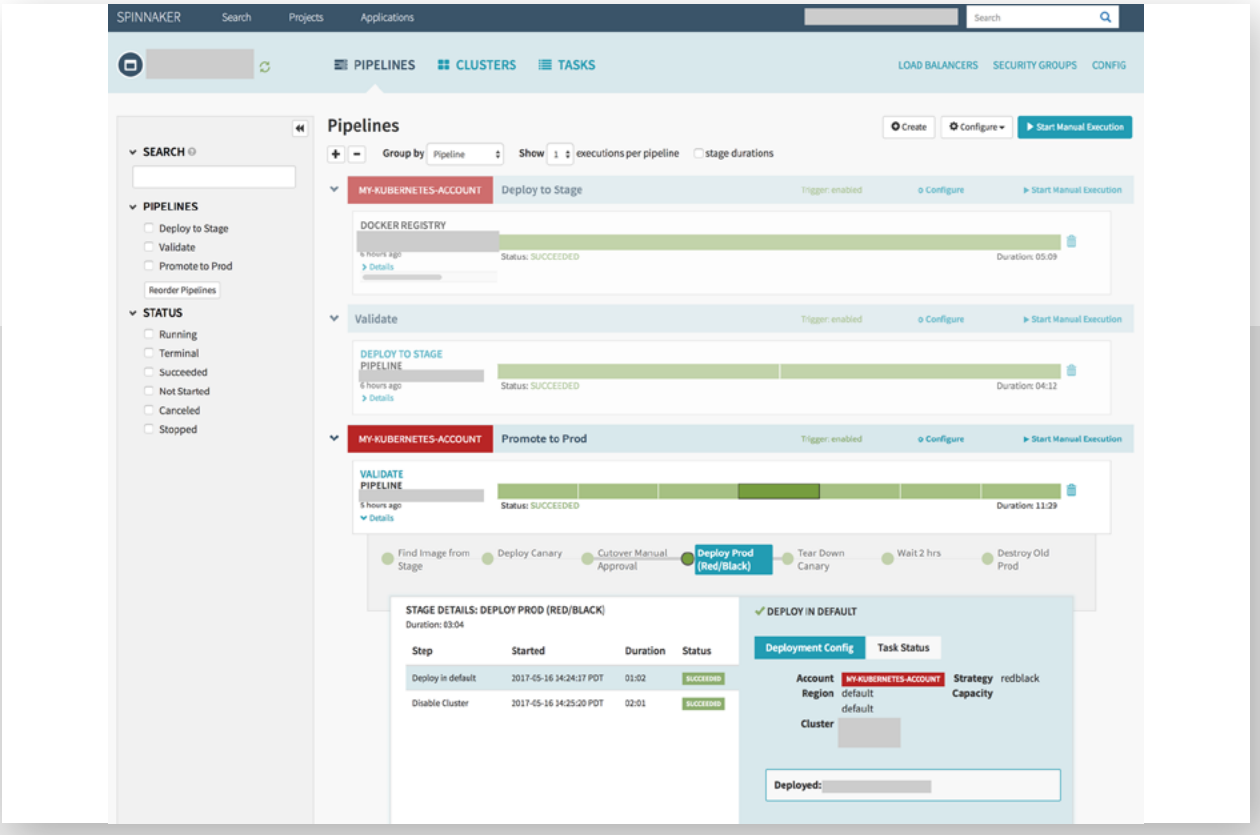
With support for a wide variety of config management tools, SSO integrations, and web hook integrations, Argo CD is a feature-rich deployment tool.

© Harness Inc. 2021



Armory Spinnaker

Armory Spinnaker is a platform built using Spinnaker at its foundation, and enhanced with some Armory modules to extend its functionality. Spinnaker, a CD tool originally built by Netflix, is an open-source tool and is reviewed later on in this guide. Armory Spinnaker is a paid product, so there is access to support and training. As for the Armory modules added on top, these include observability, expanded governance, secrets management, pipelines as code, and more.



Pros

Spinnaker users often mention the lack of support with the product, and Armory is here to save the day. Enterprise support is definitely a highlight of the tool.

- 24x7 enterprise support, up-to-date with Spinnaker open-source.
- Robust governance and secrets management.
- Pipelines as code, Terraform integration, Armory Agent for Kubernetes.

Pricing

Since this is an enterprise-grade platform, we find this tool unsuitable for smaller businesses—and definitely a hard pass for personal projects! The least expensive option is Armory's Developer plan at $60,000 per year for up to 25 users, their mid-tier plan is the Enterprise plan at $100,000 per year, and the priciest option is their Managed plan, at a whopping $150,000 per year (!). Keep in mind, too, that prices increase by user count—so the figures above are the minimum possible prices. Expect to pay more.

Cons

With Armory Spinnaker being built on Spinnaker, it does tend to carry over Spinnaker's issues—some of which are how complicated the tool is and how much it costs in terms of maintenance. Armory admitted that it often takes 2–4 full-time engineers to install, operate, and configure Spinnaker, so this should come as no surprise.

- Lacks flexibility, has underlying issues from Spinnaker.
- Expensive solution, unsure if the price is worth the extensibility from the open-source software.
- Extremely complicated, high barrier to entry, requires lots of maintenance.

—

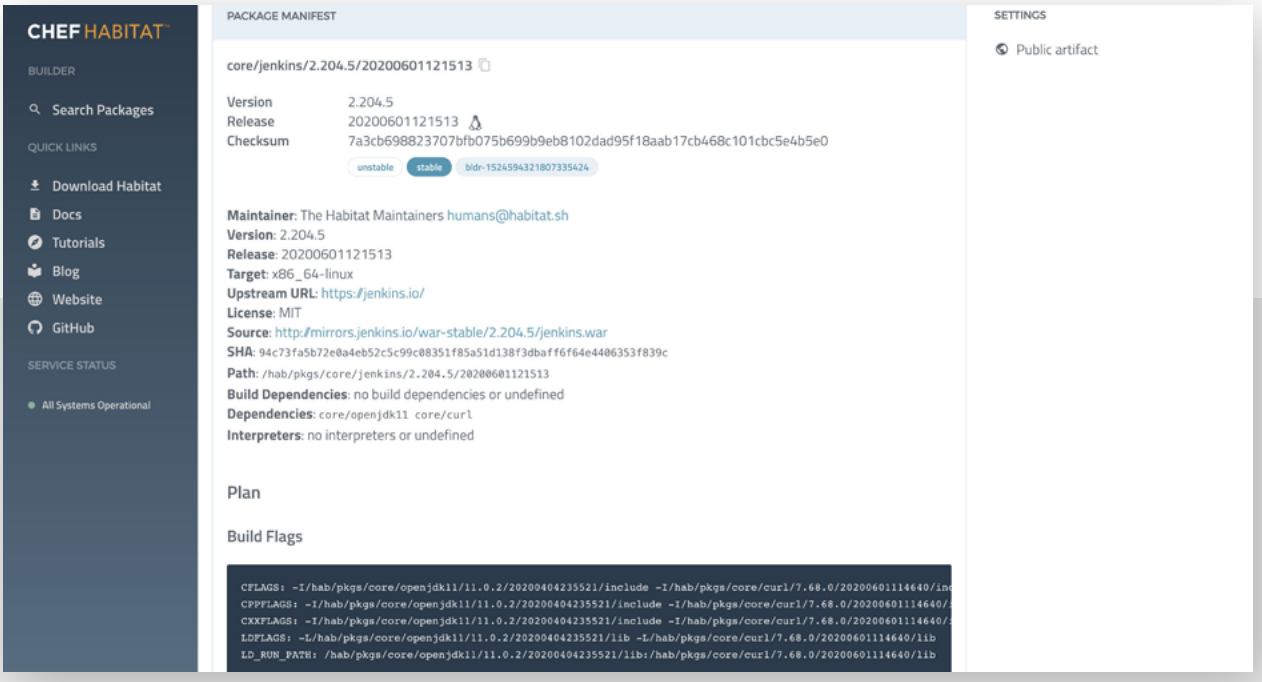
Armory admitted that it often takes 2–4 full-time engineers to install, operate, and configure Spinnaker.

© Harness Inc. 2021



Chef

Chef, a configuration management tool which is now part of Progress, is an established name in the DevOps space. They provide a plethora of automation tools known as the Chef Enterprise Automation Stack. These include tools for security, infrastructure, compliance, build and deployment, observability, and more. The whole purpose of the Chef stack is to deliver with speed, minimize risk, and maximize efficiency—all through a DevSecOps approach. It's not a CD tool in and of itself, but it does it through their Habitat module.



Pros

Chef is a very efficient solution. While not a CD tool itself, it has the module to get it done—quickly, securely, and efficiently.

- Automation—efficiency and velocity are at its core.
- Scalable and secure.
- Powerful configuration management tool backed by version control.

Cons

Lacks many desirable CD features, as it is not a CD tool in itself. Don't expect support for many integrations or languages.

- Difficult setup, docs are limited.
- Master node can only be configured in Linux/Unix.
- No YAML, and if you don't know Ruby, expect a steep learning curve.

Pricing

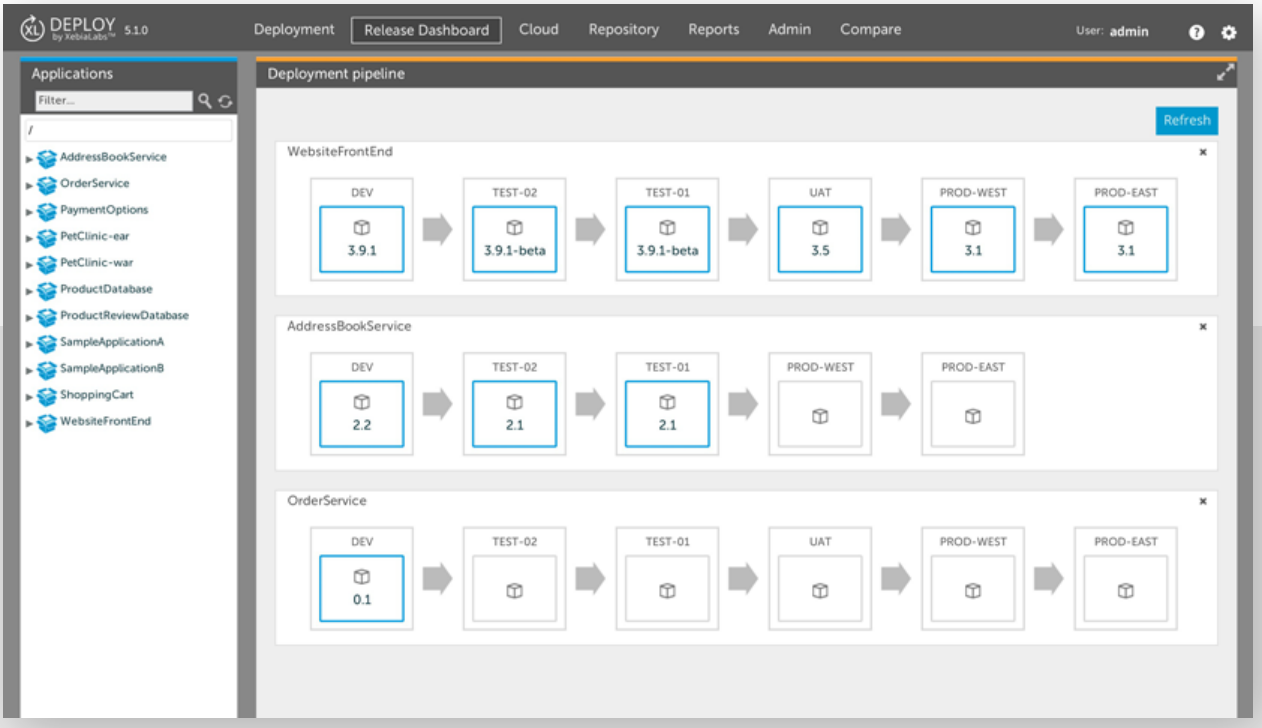
Chef recently revamped their pricing structure and is now only available after a sales consultation, but to give you an idea, the smallest plan they had prior to the revamp was for $35,000 per year for the full automation stack, while the enterprise version cost $150,000 per year. Pricing was based on nodes, targets, and service instances consumed.

> Lacks many desirable CD features, as it is not a CD tool in itself. Don't expect support for many integrations or languages.

© Harness Inc. 2021

Digital.ai Deploy

Digital.ai, formerly XebiaLabs, was featured in The Forrester Wave™: Continuous Delivery & Release Automation Q2 2020 report, where they received the highest scores possible in 19 criteria, including product innovation, security, and user experience. Their Digital.ai Deploy tool offers the ability to automate and standardize complex deployments in a highly scalable manner, allowing for a self-service environment for DevOps teams and engineers.



Pros

Digital.ai Deploy provides highly sought-after governance and security features. It's definitely an option for organizations that depend on governance and compliance capabilities for their regulated fields.

- Good governance controls, such as RBAC and audit capabilities.
- Quick rollbacks, self-service, secure.
- Tons of integrations, supports containers.

Cons

The biggest con to Digital.ai Deploy is the sheer amount of engineering time it requires to set up, configure, and maintain. This type of toil is understood when dealing with open-source tools, but for a paid product to require so many resources on top of it is rather off-putting.

- Documentation is subpar.
- Requires a hefty amount of time in maintenance and configuration.
- Steep learning curve, difficult setup.

Pricing

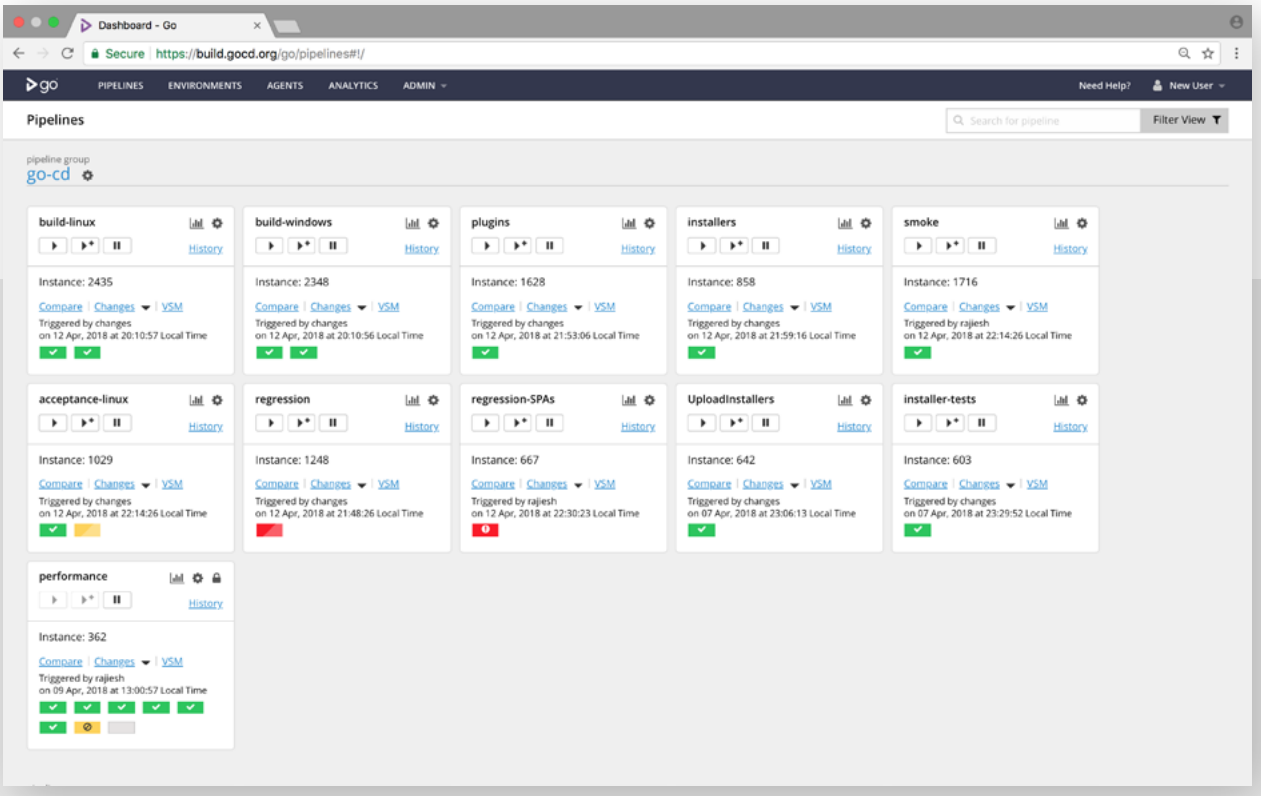
No public pricing information available, but there is information on their DFP program that can explain some of the costs.

The biggest con to Digital.ai Deploy is the sheer amount of engineering time it requires to set up, configure, and maintain.

© Harness Inc. 2021

GoCD

GoCD is an open-source tool sponsored by ThoughtWorks, a global software consultancy firm. This tool boasts a plethora of features, such as pipelines as code, a native Kubernetes integration, elastic agents, JSON and YAML support, and a modern interface. It is, however, a newer tool—as such, experienced users may not like the lack of more advanced features. Otherwise, there are a healthy amount of plugins available—nothing overwhelming like Jenkins, but enough to enhance the extensibility of the tool. At 6k stars on GitHub, over 120 contributors, and an active community, it's not a tool to scoff at.



Pros

GoCD seems to hit all the hot buzzwords right now: pipelines as code, Kubernetes, and GitOps—oh my! There is also a surprisingly active community surrounding the tool, so documentation issues aren't as big of a problem on this open-source solution.

- Active community for support, good docs, reliable.
- Tons of high-demand features right out of the box.
- Runs pipelines in parallel, self-service, supports most languages or frameworks.

Pricing

Open-source strikes yet again! Do however remember the hidden costs of open-source.

Cons

The UI on GoCD isn't the best. It's very busy and users have reported difficulties in finding where you're supposed to go. It's not very intuitive as of yet.

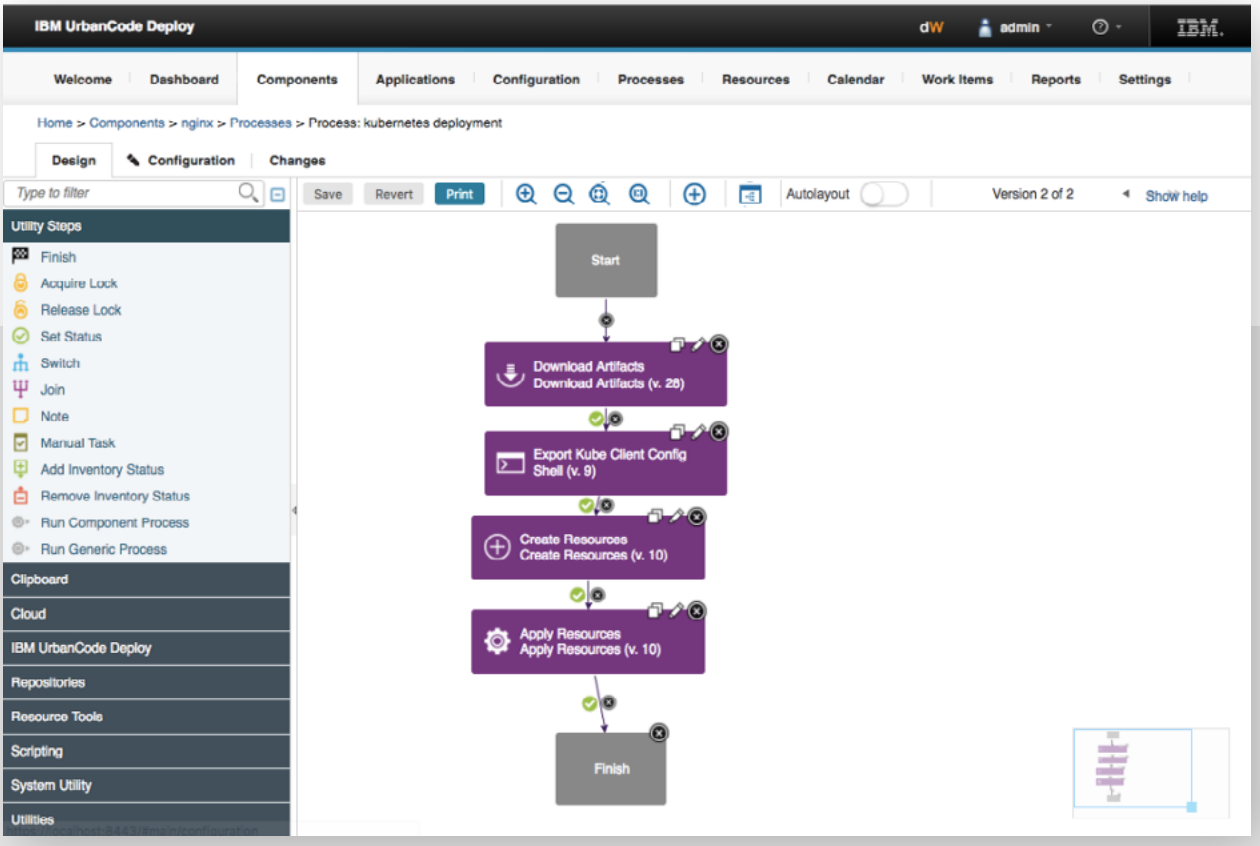
- As with any open-source software, there are often bugs to work through.
- Interface can take a bit to get used to, but improvements are actively being made.
- Single XML configuration for all pipelines.

> GoCD seems to hit all the hot buzzwords right now: pipelines as code, Kubernetes, and GitOps—oh my.

© Harness Inc. 2021

IBM Cloud DevOps

It makes sense that IBM would want to enter the DevOps game, being a huge name in tech and all. Their Cloud DevOps suite can be used to automate key steps in the deployment process, and their specific CD tool, UrbanCode Deploy, accelerates and optimizes software delivery. The tool automates deployments and rollbacks, and the provisioning, updating, and decommissioning of environments. UrbanCode Deploy boasts a heaping helping of plugins to extend the tool as well.



Pros

Some people enjoy relying on the safety, reputation, and trust of big organizations—and IBM definitely fits the bill, as a trusted name in tech for decades.

- Good integrations, UI is simple and easy to understand.
- Snapshot functionality allows access to previous versions of artifacts.
- Little downtime, though new releases cause bugs.

Cons

The solution is quite resource-intensive and simple things like running concurrent deployments can slow the tool down dramatically.

- Convoluted documentation, complex setup.
- Requires lots of resources to run, clunky, cost-prohibitive.
- Running multiple deployments slows product down.

Pricing

Pricing is based on licenses, which are based on the number of servers that are being deployed and the number of agents that are used. Users have found the product to be too expensive for what it offers.

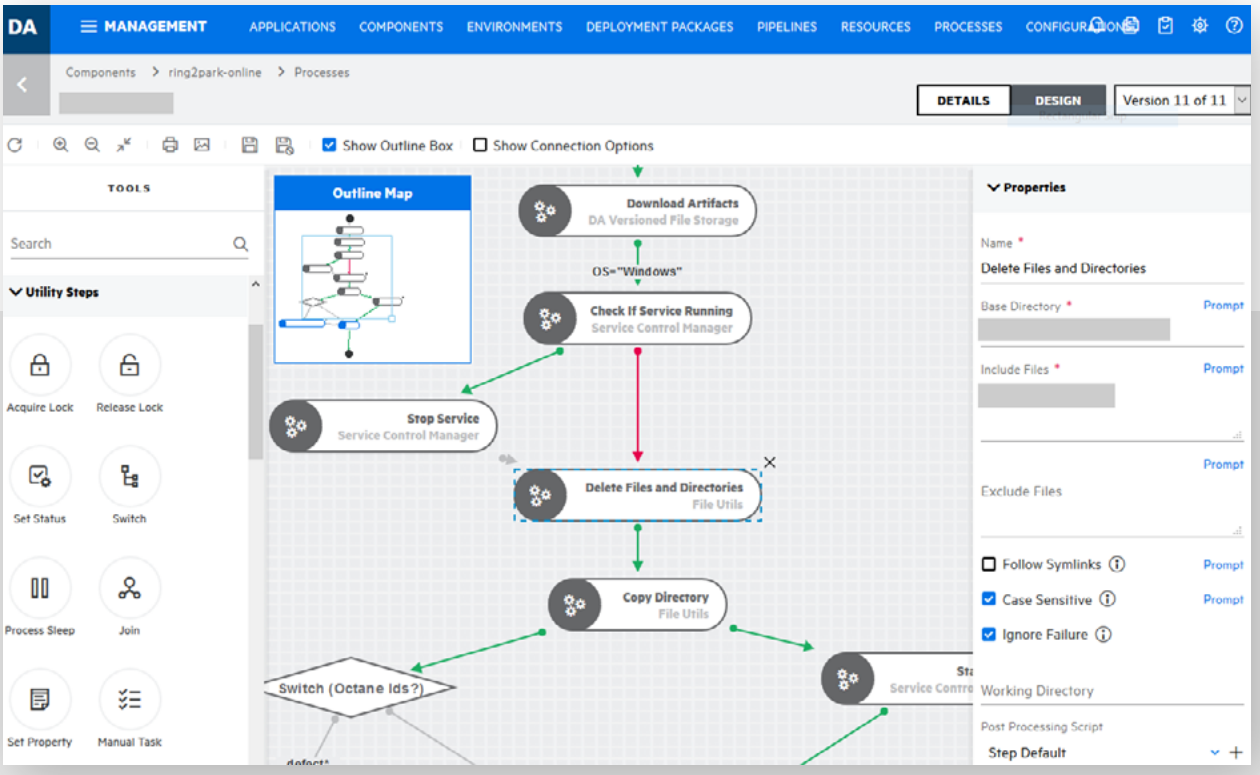
—

Some people enjoy relying on the safety, reputation, and trust of big organizations—and IBM definitely fits the bill, as a trusted name in tech for decades.

© Harness Inc. 2021

Micro Focus

Micro Focus is a digital transformation company. They help businesses modernize their processes with tools across DevOps, hybrid IT management, security, predictive analytics, and of course, continuous deployment. Their CD product, Deployment Automation, offers drag-and-drop simplicity, deployment automation, deployment pipeline models and management, and CI/CD integrations.



Pros

Since Deployment Automation was acquired by Micro Focus, users have noticed a dramatic improvement to the product's support department, and with the overall look and feel of the tool.

- Replaces the need for scripting, has good plugin library.
- Rapid adoption by engineers once product is live, intuitive UI.
- Multiple support options, how-to videos, community.

Cons

The tool still doesn't provide much in terms of third-party integrations. If your current tool utilizes lots of plugins and third-party vendor tools, Deployment Automation isn't for you.

- Initial migration to platform can cause issues and take months.
- Few third-party integrations available.
- Rigid contract, few capabilities.

Pricing

No public pricing information available. Users have found the product to be too expensive for what it offers.

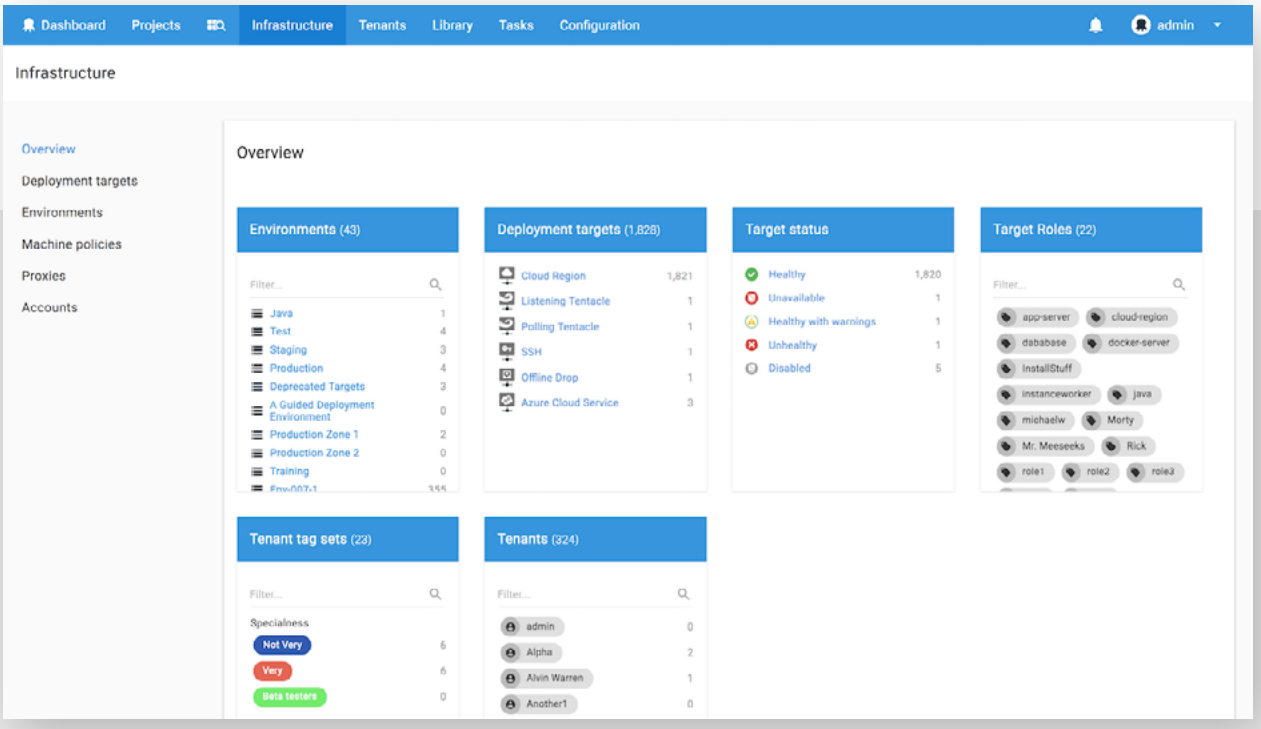
Since Deployment Automation was acquired by Micro Focus, users have noticed a dramatic improvement to the product's support department, and with the overall look and feel of the tool.

© Harness Inc. 2021



Octopus Deploy

Octopus Deploy is a CD tool that manages releases, automates deployments, and helps you ship code faster. It does have some nice features: lots of deployment strategies, good support, and integrates with most CI servers out there. However, many of the features are quite basic —which we suppose might make it more attractive for organizations that are new to CI/CD. For instance, there's no YAML in Octopus Deploy—this might be appealing for newbies, but disenchanting for advanced engineers and DevOps teams that like to write their own config as code. The tool simply doesn't have the kind of functionality that advanced CD users require. On the plus side, there are updates coming that should better position Octopus Deploy as a true player in the CD space.



Pros

For a new organization just starting out, the simplicity of Octopus can be a good thing: low barrier to entry, plenty of options on just about everything—there's no rigidity in that you must know this specific language or deployment strategy in order to use the tool.

- Multiple deployment strategies available—blue/green, rolling, or canary.
- Extensive integrations, language support, operating systems, platforms, etc.
- Good docs, responsive support, regular updates.

Cons

The simplicity stated above will be largely off-putting for advanced users. As flexible as Octopus is, it lacks advanced features and even ones we consider 'basic' (meaning: necessary) like observability.

- No observability—lacks useful reports and metrics.
- Initial setup and configuration is complex.
- Pipeline configuration is not intuitive when configuring deployments across multiple environments.

Pricing

Depends on where it's hosted. Both versions are free for 10 targets or less. Pricing incrementally increases after that figure—may we add, exponentially on the on-prem version—which is perhaps baffling.

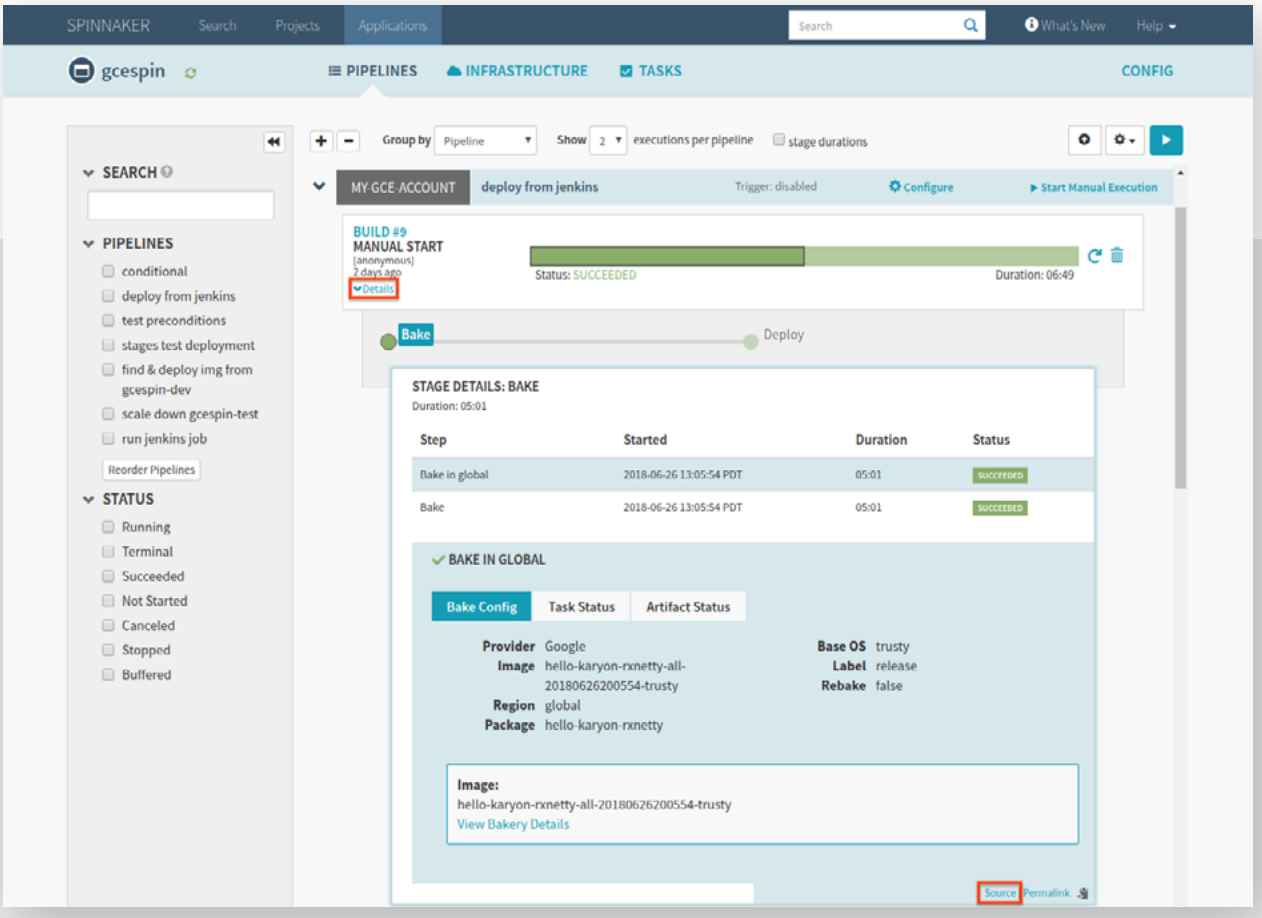
The tool simply doesn't have the kind of functionality that advanced CD users require. On the plus side, there are updates coming that should better position Octopus Deploy as a true player in the CD space.

© Harness Inc. 2021



Spinnaker

Spinnaker is an open-source continuous delivery platform that was originally created by the engineers at Netflix. They open-sourced it back in 2015, and as of now, it has 7.6k stars on GitHub, 116 contributors... and over 300 open issues. That's not to say Spinnaker doesn't have its perks—it's flexible, reverts deployments easily, and provides good auditing capabilities. However, we've heard from countless users that managing Spinnaker is a full-time job by itself; the overhead simply makes this solution unsuitable. The good news is you can easily see for yourself—the tool is free, so it's not like you'd be making a pricy investment into something that may not work out. Just remember that the investment is in time—time that could be used to build more business functionality.



Pros

Spinnaker, an open-source tool, is consistently maintained and improved. New features and bug fixes routinely become available, and it does offer sought-after features.

- Very customizable pipelines, flexible.
- Automated canary analysis.
- Built-in support for lots of monitoring and observability tools.

Cons

Managing Spinnaker requires between 2–4 FTEs. This makes it almost as complex and expensive as Jenkins, which we find unreasonable when it comes to open-source software.

- Busy UI, steep learning curve.
- Initial setup and configuration is an absolute nightmare.
- Support is spotty at best, bugs can take a long time to fix.

Pricing

Open-source! No pricing here, but do remember the hidden costs of open-source software when making your decision.

—

We've heard from countless users that managing Spinnaker is a full-time job by itself; the overhead simply makes this solution unsuitable.

© Harness Inc. 2021

03
CI/CD Tools

Key

At Harness, we've found that most CI/CD challenges come from the below four pain points. We've built our buyer's guide with these in mind. The icons will act as a key for all reviewed vendors. Look to the right of the vendor's name to see which pain points the solution most addresses.

Velocity
Time-to-Market
Developer Velocity & Onboarding
Cloud & Microservices Migration

Quality
Downtime & Incidents
Defects & Regressions
Performance

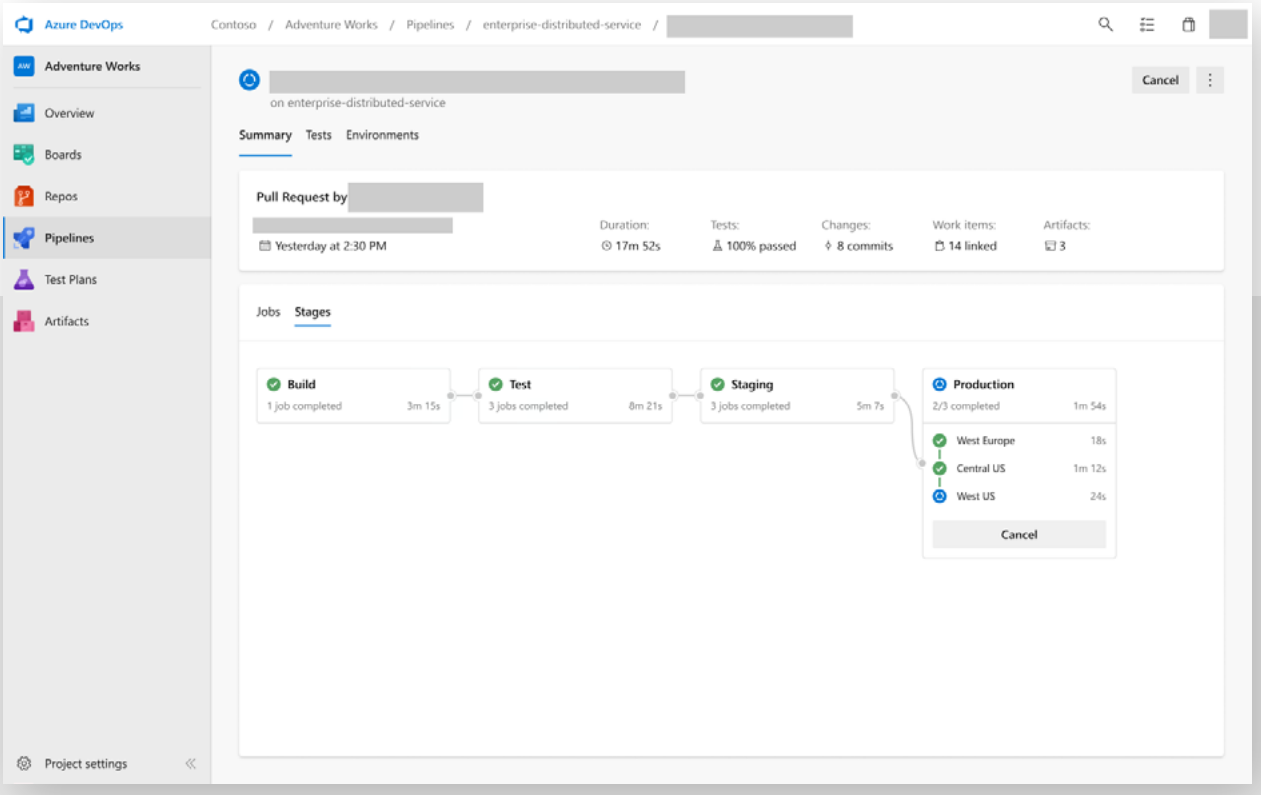
Governance
Standardized Deployments
Auditing & Compliance
Role-Based Access Control

Efficiency
Developer Productivity
Cloud Costs

© Harness Inc. 2021

Azure DevOps

This platform has many different modules consisting of a code repository, CI/CD pipelines (Azure Pipelines), artifact management, community extensions, test tools, and Kanban boards. As expected from Microsoft products, Azure DevOps offers fine-grained RBAC, audit trails, and deployment verification with rollbacks. It works well with cloud-native technologies and has support for legacy technologies. Azure DevOps is offered as a SaaS solution but also has an on-prem version called Azure DevOps Server.



Pros

For businesses that like having the trusted Microsoft name—and who happen to be using Azure already as their cloud provider—Azure DevOps can make sense as a CI/CD provider. Especially if their business is in a heavily restricted or regulated field.

- Some of the finest governance capabilities among CI/CD vendors.
- Pipelines are very flexible and customizable.
- Great Git capabilities since Microsoft acquired GitHub.

Cons

Azure DevOps can be hard to navigate initially—users have reported difficulty getting around and finding what they need. Documentation is fairly good, but lacking in some areas and may present difficulties when trying to solve more obscure and complex issues.

- UI looks great but it can be hard to find what you need.
- Manual automatic deployment verification, some scripting required.
- Documentation is lacking in certain areas.

Pricing

Azure changes monthly per user license, with the first 5 on the Basic plan being free, then $6 per user per month. The Basic + Test plan (way more feature-rich) is $52 per user per month. They have a handy pricing sheet where you can check on user license needs.

For businesses that like having the trusted Microsoft name—and who happen to be using Azure already as their cloud provider—Azure DevOps can make sense as a CI/CD provider. Especially if their business is in a heavily restricted or regulated field.

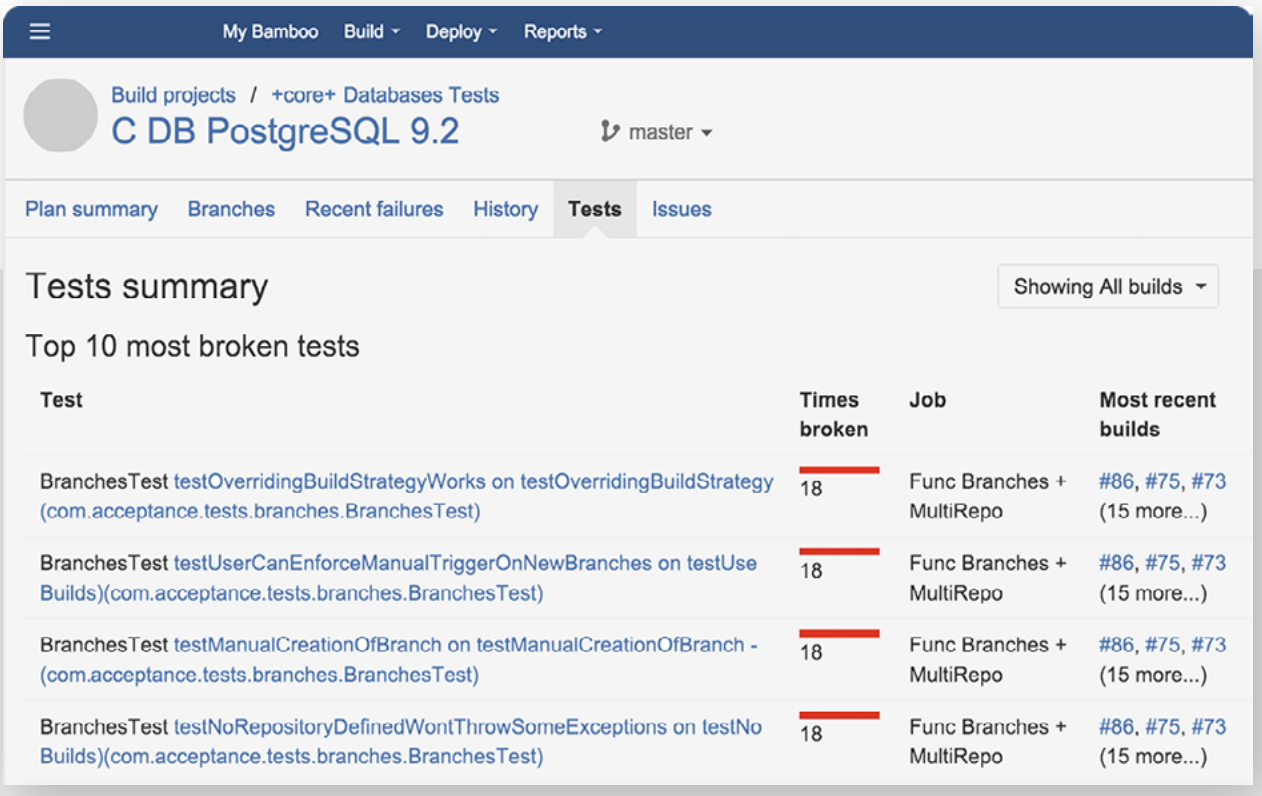
© Harness Inc. 2021



Bamboo



Bamboo, by Atlassian, is a CI/CD tool marketed to professional teams. Atlassian has great tools, namely JIRA and Bitbucket, but they missed the mark slightly with Bamboo. It's a good basic solution—but if you're looking for something more feature-rich, with, for example, cloud cost optimization and monitoring/observability, you'll need to look for a different tool. That's not to say there aren't good things about it: it supports pipelines as code, is suitable for any language, and as we stated, integrates super well with the rest of the Atlassian suite. It just won't perform to the level of some of the other CI/CD platforms reviewed in this guide.



Pros

If your business uses JIRA and Bitbucket already, choosing Bamboo as your CI/CD tool might make sense for the flawless integration and the 'everything under one roof' preference some companies have.

- Atlassian user? You'll love the integrations to JIRA/Bitbucket.
- Fairly simple UI, fast builds.
- Decent reporting and auditing capabilities.

Pricing

Bamboo offers two pricing plans. Their Small Teams plan provides up to 10 jobs for $10 per month, and all proceeds go to charity, which is a really sweet way to give back to the community. The Growing Teams plan is exponentially more pricey, especially as you get up there in remote agents (that's how the product bills you—by remote agents, not by user licenses, like many other CI/CD vendors do).

Cons

Something about this tool was a bit shocking to us: the plugin marketplace actually sells plugins. We're used to plugins being free, so seeing the price tag on some of these was jarring—especially since Bamboo is a paid product, not an open-source tool.

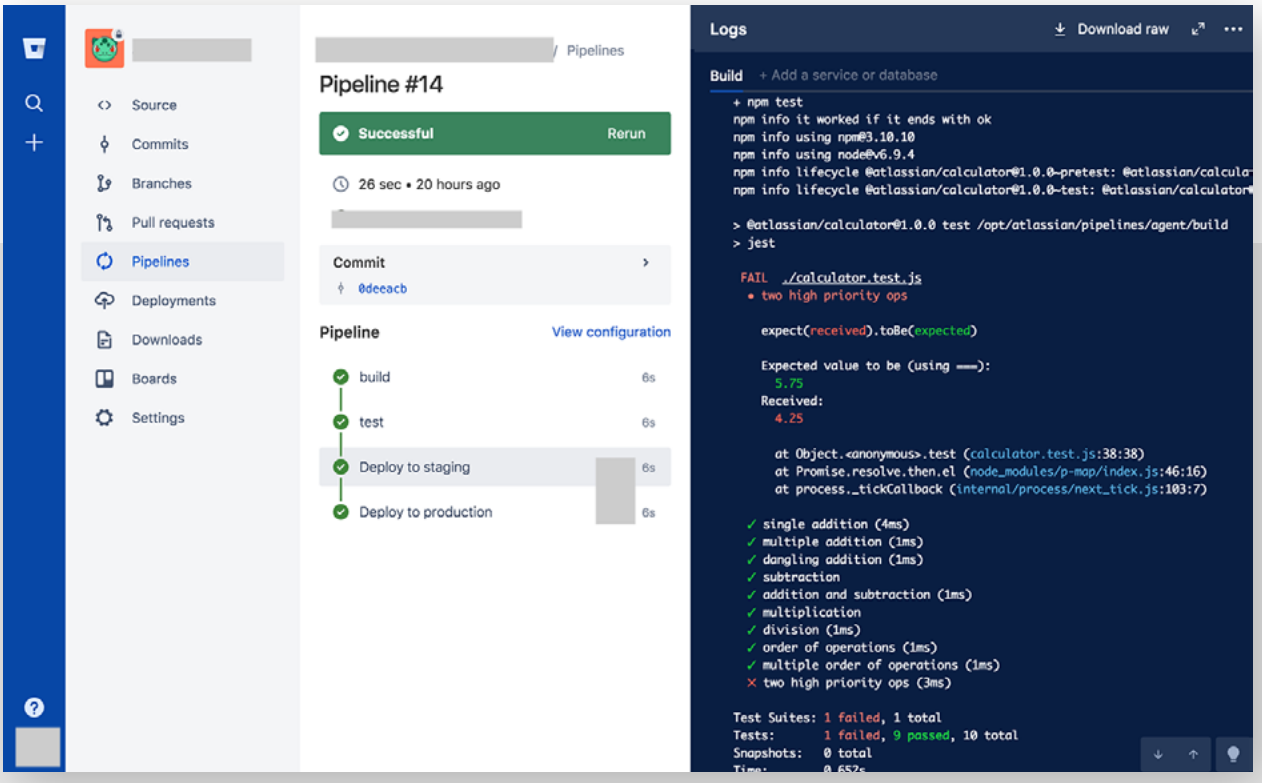
- Depends on plugins for extensibility, which results in lots of overhead in time and money.
- Resource-intensive, no cloud solution, limited docs and support.
- Value seems somewhat dependent on use of other Atlassian products.

Something about this tool was a bit shocking to us: the plugin marketplace actually sells plugins. We're used to plugins being free, so seeing the price tag on some of these was jarring—especially since Bamboo is a paid product, not an open-source tool.

© Harness Inc. 2021

Bitbucket Pipelines

Bitbucket is, funnily enough, another Atlassian product. It started off as a Git solution, but in 2020, they added integrated CI/CD functionality—because what's easier than CI/CD directly inside your source code management tool? As with Bamboo, a big part of the value of Atlassian's tools is their integration with each other—so lovers of JIRA might find Bamboo or Bitbucket Pipelines useful. An observation, which will be obvious: Bitbucket Pipelines will not be useful if you use another Git tool, like GitHub or GitLab—only if you already use Bitbucket or are planning to move to it. Lastly, the tool was created to make life easier—so the functionality of it is a little basic; it's basically build, test, deploy, done. Good beginner CI/CD tool.



Pros

If your business uses JIRA and Bitbucket already, choosing Bitbucket Pipelines as your CI/CD tool might make sense for the flawless integration and the 'everything under one roof' preference some companies have.

- Easy setup done in two steps, not a ton of maintenance or management.
- Great integration with other Atlassian tools and with a ton of third-party tools.
- Exceptional visibility into pipelines/builds and collaboration with teams.

Cons

This is a beginner tool lacking in most advanced features. You can create basic pipelines quite easily, so this would be a good solution for a personal project and perhaps small businesses with simple CI/CD needs, but bigger organizations/projects should avoid this solution.

- Good for basic pipelines but more complex projects might require a more feature-rich platform.
- No governance, no automatic deployment verification.
- System sometimes produces odd errors that require lots of troubleshooting.

Pricing

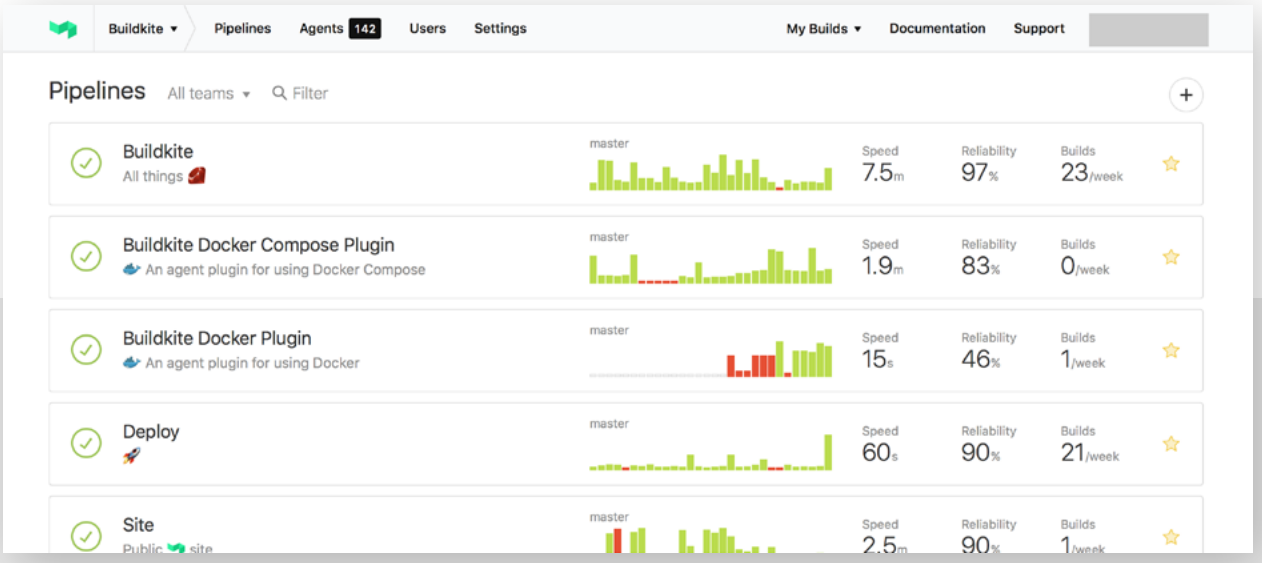
Priced by users per month. Teams of 5 or less are free, while the Standard plan is $3 per user per month, and the Premium plan is $6 per user per month. To see a full comparison between plans, visit the pricing page. It's important to note that there is a cap on build minutes on each plan, but additional minutes can be purchased for $10/1000 minutes.

Bitbucket Pipelines will not be useful if you use another Git tool, like GitHub or GitLab —only if you already use Bitbucket or are planning to move to it.

© Harness Inc. 2021

Buildkite

Buildkite is a newer player in the CI/CD area. It started off as a CI tool but has added functionality to handle CD as well. As a smaller company, Buildkite relies heavily on plugins to extend the platform's capabilities. Pipelines are fast and easily scalable, basically any language is supported, and there are tons of integrations. However, the platform is still fairly new and lacks much of the modern features more mature platforms boast.



Pros

For a new tool, it is furnished with some good features, like audit trails and scalability. The UI is also quite beautiful and intuitive.

- Easy setup, simple UI, easy to maintain.
- Integrates well with source code management tools.
- Easy to scale up and down, good logging, very customizable.

Pricing

Like most platforms, Buildkite has a free plan. Their Standard plan is $15 per user per month, with an upper limit of 200 users. Their Enterprise plan is $2999 per month up to 100 users, with a $29 per month fee for additional users.

Cons

It's a new tool, which means little documentation and community support. Companies in rigid/regulated industries should stay away, as the lack of governance and compliance features makes Buildkite a nonstarter.

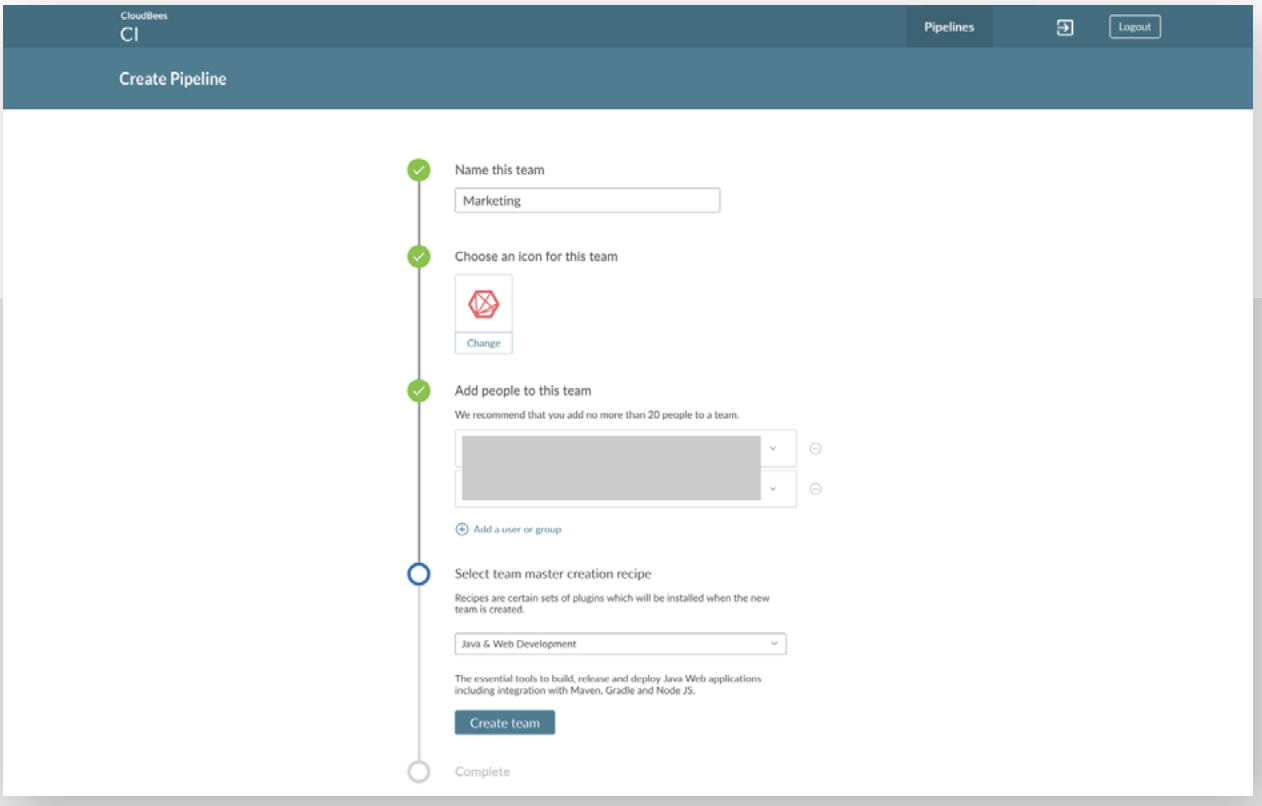
- Newer product, so lack of community support.
- Lacks some modern features.
- Not much in terms of governance, secrets management is doable but requires plugins or third party tools.

> Pipelines are fast and easily scalable, basically any language is supported, and there are tons of integrations.

© Harness Inc. 2021

CloudBees

CloudBees is basically the enterprise version of Jenkins, which means it unfortunately carries over the problems Jenkins has: scripting, plugins, dependency hell—it's all there. Plugins aren't containerized (unlike other solutions that are container-native), and toil lost on maintenance is still a problem. However, this also means that it does offer the good of Jenkins as well: extensibility, integrations, and tons of support. On the CD side, CloudBees offers decent governance features, such as RBAC and audit trails—but there are a lot of features that are missing.



Pros

CloudBees is built on Jenkins, which means it gets all the advantages of Jenkins: extensibility with over 1,800 plugins, customizable, and able to do pretty much anything you need it to with scripting.

- Platform is extensible and very customizable, powerful plugins available.
- More robust than Jenkins on its own.
- Much of the value is built upon CloudBees' support.

Cons

There's not that much difference between CloudBees and Jenkins, except for the support that CloudBees provides. If you don't depend much on support for CI/CD, it may not be worth it to go with a solution that bases most of its value on its support. Additionally, it comes with the good of Jenkins, but also the bad—so keep in mind the scripting toil and maintenance hell that can come with it.

- Changes and bug fixes are sporadic.
- Not cloud-native, lacks API compatibility and continuous verification capabilities.
- Old UI, logs are hard to find, tons of toil/maintenance.

Pricing

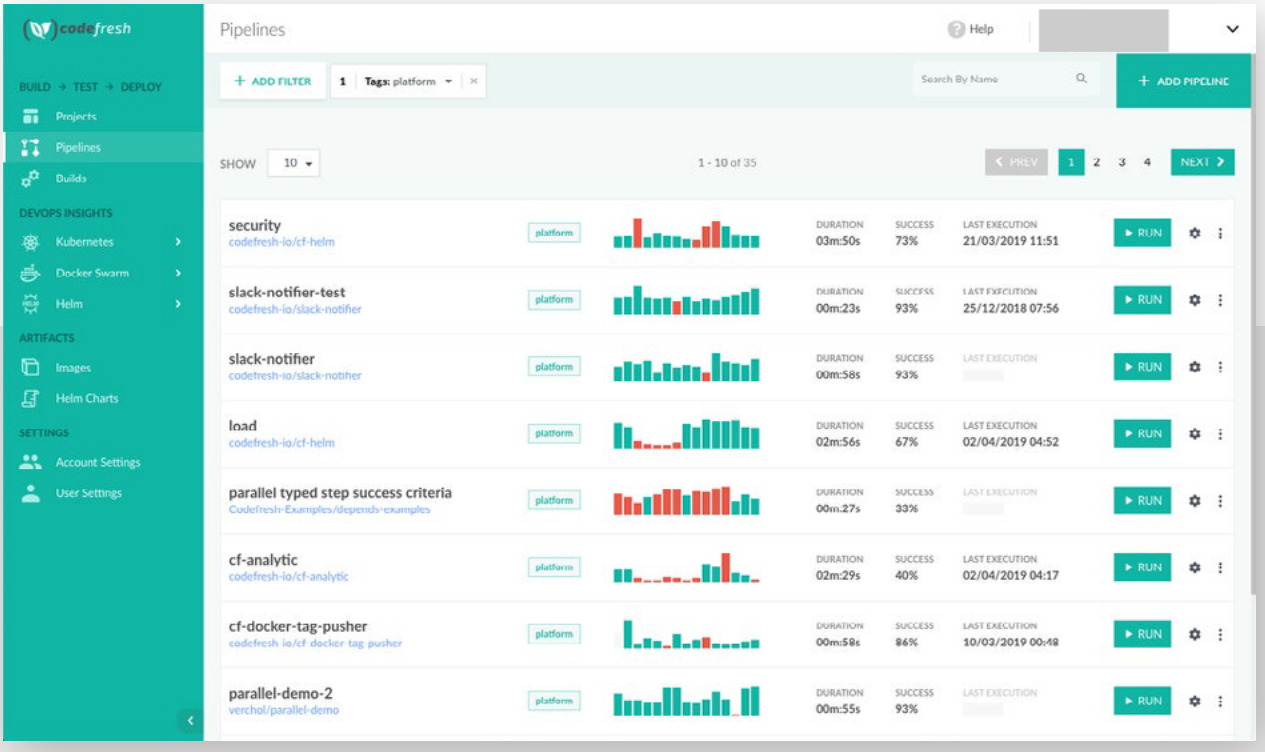
No public pricing information available. Users have found the product to be too expensive for what it offers.

CloudBees is basically the enterprise version of Jenkins, which means it unfortunately carries over the problems Jenkins has: scripting, plugins, dependency hell—it's all there.

© Harness Inc. 2021

Codefresh

Codefresh brings CI/CD, Kubernetes, and GitOps together into a beautiful UI. A modern, feature-rich, cloud-native, container-based platform, this one knocks it out of the park in many categories. It has strong governance features, little downtime, integrates with just about anything, and creates very flexible pipelines. It also supports Linux, Windows, Arm, and Mac OS and boasts multiple deployment strategies. Codefresh offers end-to-end CI/CD capabilities.



Pros

Codefresh boasts many desirable features and integrations. It's fairly robust when it comes to governance, making it a good choice for heavily-regulated industries.

- Oodles of integrations, cloud-native, container-based, pre-built pipeline templates.
- Beautiful UI, super fast, scalable.
- Governance focus: SAML SSO, audit trails, RBAC.

Cons

As a newer tool, there are occasional bugs, stability issues, and some downtime—though, not much on the latter. The biggest concern is in terms of resources. Cloud costs can come up quite a bit on this platform, so keeping a watchful eye on resource consumption will be necessary.

- Consumes a lot of resources—easy to run out on your plan.
- Newer products so docs/community support needs improvement.
- Bugs are encountered once in a while, as are stability issues.

Pricing

Codefresh prices by instance size and concurrent builds. They offer a free, very basic plan with one small instance and one concurrent build. Otherwise, you'll have to choose between the Basic, Pro, or Enterprise plans. Depending on your needs, it could get pricey quickly. See the full pricing structure on their pricing page.

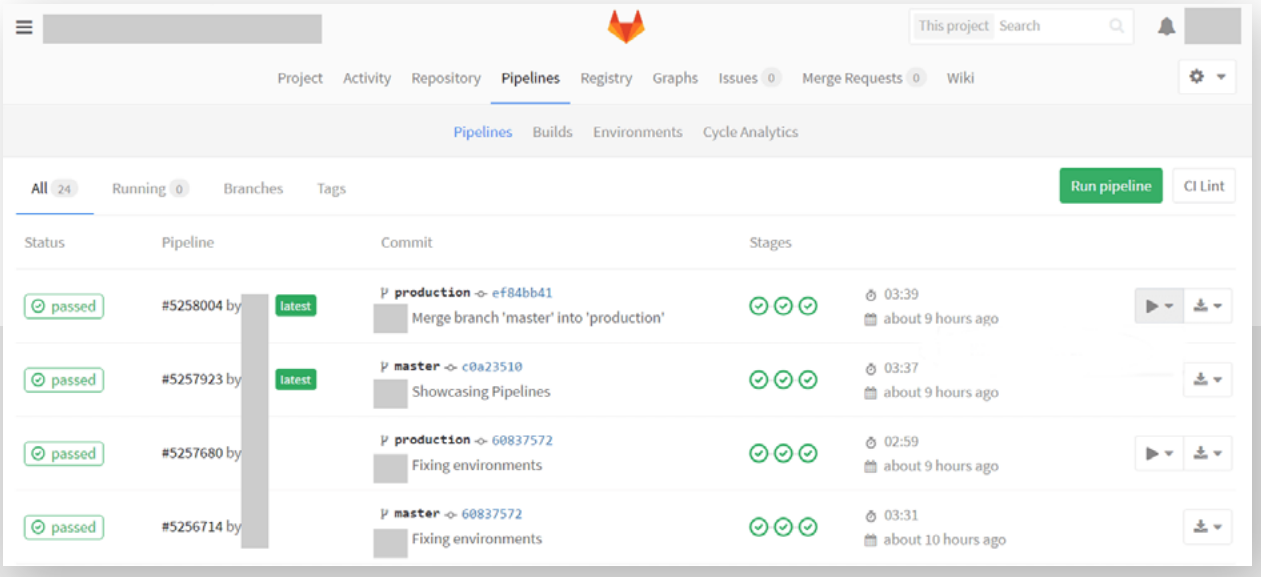
—

A modern, feature-rich, cloud-native, container-based platform, this one knocks it out of the park in many categories.

© Harness Inc. 2021

GitLab

GitLab is a bit different from other CI/CD platforms listed here since it is also a popular code repository (like Bitbucket Pipelines, actually). GitLab offers robust Continuous Integration, reliable Continuous Deployment, manual deployment verification and rollback, application security testing, and issue tracking/planning. As for governance, GitLab is a little limited in that area. It can be managed with certain third-party integrations, but lacks native functionality.



Pros

Combining source code management and CI/CD can make life simpler—having everything under one roof is a preference for many companies, and GitLab can be a good choice for that purpose.

- CI/CD and repos in one place on one of the best Git platforms.
- Fast, scalable, supports multiple platforms and languages.
- Real-time logging, flexible pipelines, YAML, canary deployments.

Cons

GitLab's deployments lack many features advanced CI/CD users have come to expect, like automatic rollbacks and deployment verification. It's also not a good choice for companies that need solid governance features.

- Manual verification, rollbacks, and Canary deployments.
- Bit of a learning curve in finding where everything is and how it works.
- RBAC is only supported with GitLab's native Kubernetes integration—other governance features are lacking.

Pricing

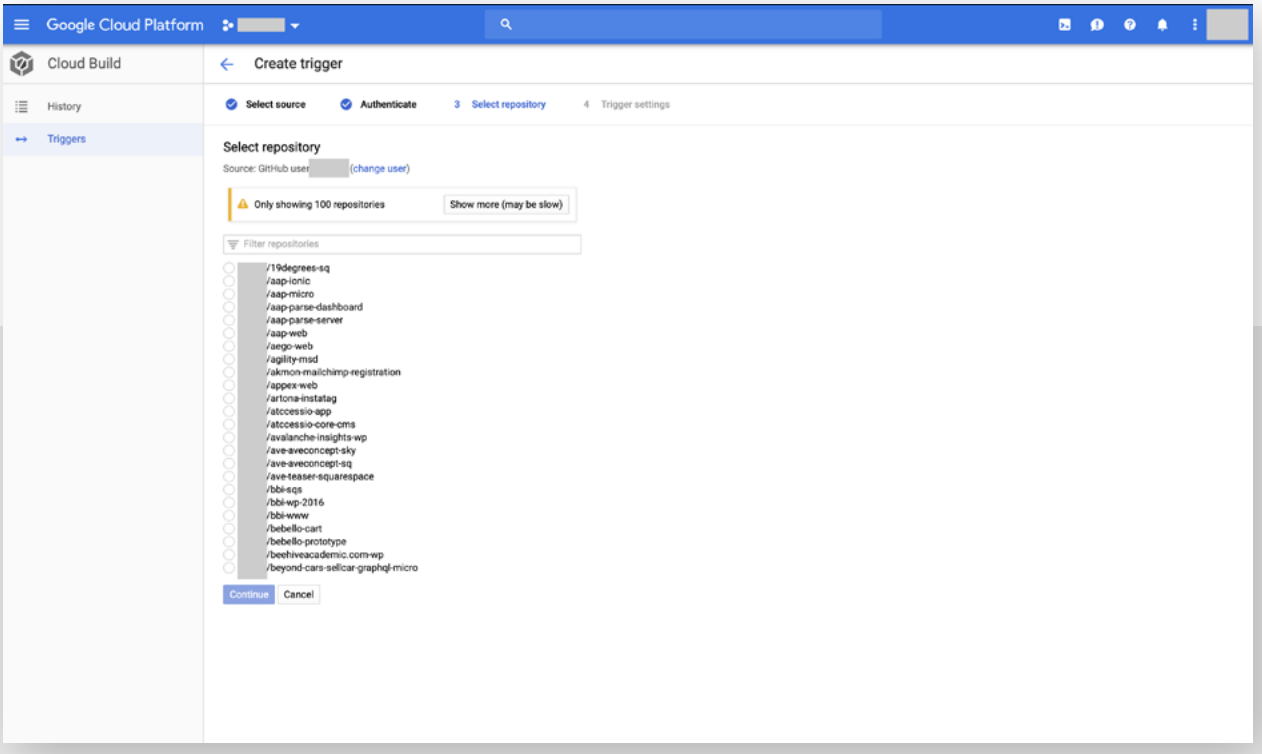
GitLab charges per user per month, along with a cap on CI/CD minutes. As with all major CI/CD vendors, there is a free plan (very basic, for personal projects and perhaps small businesses, capping out at 400 CI/CD minutes). The Premium plan is $19 per user per month with 10,000 CI/CD minutes, and the Ultimate plan ends up at $99 per user per month with 50,000 CI/CD minutes.

Combining source code management and CI/CD can make life simpler—having everything under one roof is a preference for many companies, and GitLab can be a good choice for that purpose.

© Harness Inc. 2021

Google Cloud Build

Cloud Build, Google Cloud's CI/CD platform, was named a Leader in The Forrester Wave™: Cloud-Native Continuous Integration Tools, Q3 2019, receiving top scores in the current offering and strategy categories. This serverless CI/CD platform allows you to build quickly across all languages, and across multiple environments, such as VMs, serverless, Kubernetes, or Firebase.



Pros

Google Cloud Platform (GCP) is one of the three big cloud providers (alongside Microsoft Azure and Amazon Web Services). If you enjoy the concept of using as few vendors as possible and your cloud provider is Google, consider Google Cloud Build as your CI/CD tool. It boasts modern features and a generous plan.

- Docker-based architecture, flexible build process—use pre-built templates or build your own.
- Fast builds, profound insights, lots of YAML config file examples and templates available.
- Generous free minutes, on-prem or SaaS, good logging.

Cons

On par with other Google products, support is convoluted and the learning curve is steep. Users report general confusion and a poor UI.

- Learning curve: can take a few builds to get really comfortable.
- Google support is confusing, can take multiple reps to speak to the right one.
- UI not as nice as many other CI/CD platforms.

Pricing

Google Cloud Build charges by the build-minute, with the first 120 build-minutes per day being free. Costs are calculated based on machine type, vCPUs, and Memory. The pricing structure is very similar to AWS CodeBuild, a CI tool reviewed earlier.

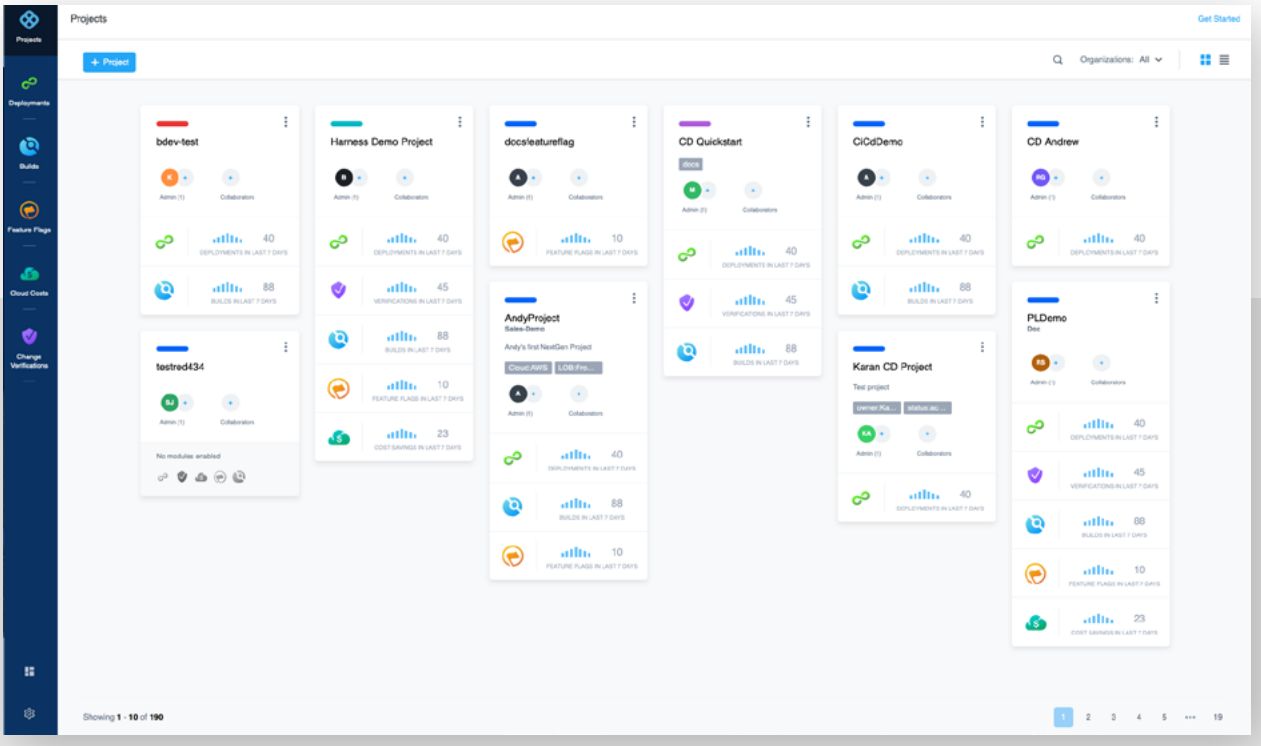
—

Cloud Build was named a Leader in The Forrester Wave™: Cloud-Native Continuous Integration Tools, Q3 2019, receiving top scores in the current offering and strategy categories.

© Harness Inc. 2021

Harness

Harness is a modern CI/CD platform that offers Continuous Integration, Continuous Deployment, Cloud Cost Management, Continuous Verification, and Feature Flag Management. Harness leads the pack when it comes to governance, with fine-grained RBAC, full audit trails, and proprietary/integrated secrets management. It was built for cloud-native technologies but also works well with legacy tools. Harness is available as a SaaS platform and on-prem for companies that require it.



Pros

Cloud Cost Management, one of Harness' modules, ensures your cloud bill is never a surprise with its anomaly detection, What If analysis, and more. Continuous Verification checks your deployments for anomalies and automatically rolls back to the previous stable version. Additionally, the platform itself is extremely feature-rich. It is suitable for every use case, and also has a strong focus on governance.

- True platform with CI, CD, CCM, CV, and FFM—feature-rich and constantly updated.
- Strong focus on governance—RBAC, SSO, audit trails, secrets management, and more.
- Cloud-native, gorgeous and easy-to-use UI, canary deployments, AI/ML-powered.

Pricing

Unit pricing is based on service instances. Each instance costs $300, but there are bulk/volume percentage discounts available.

Cons

As with most platforms, there is a learning curve. Simple pipelines can be created quickly, but more complex setups will take more time. However, the platform becomes second-nature after familiarizing yourself with it.

- Limited support for technologies outside of swimlanes.
- Initial setup has learning curve but simple pipelines can be created in 15 minutes.
- Sticker shock; but ROI achieved quickly.

—

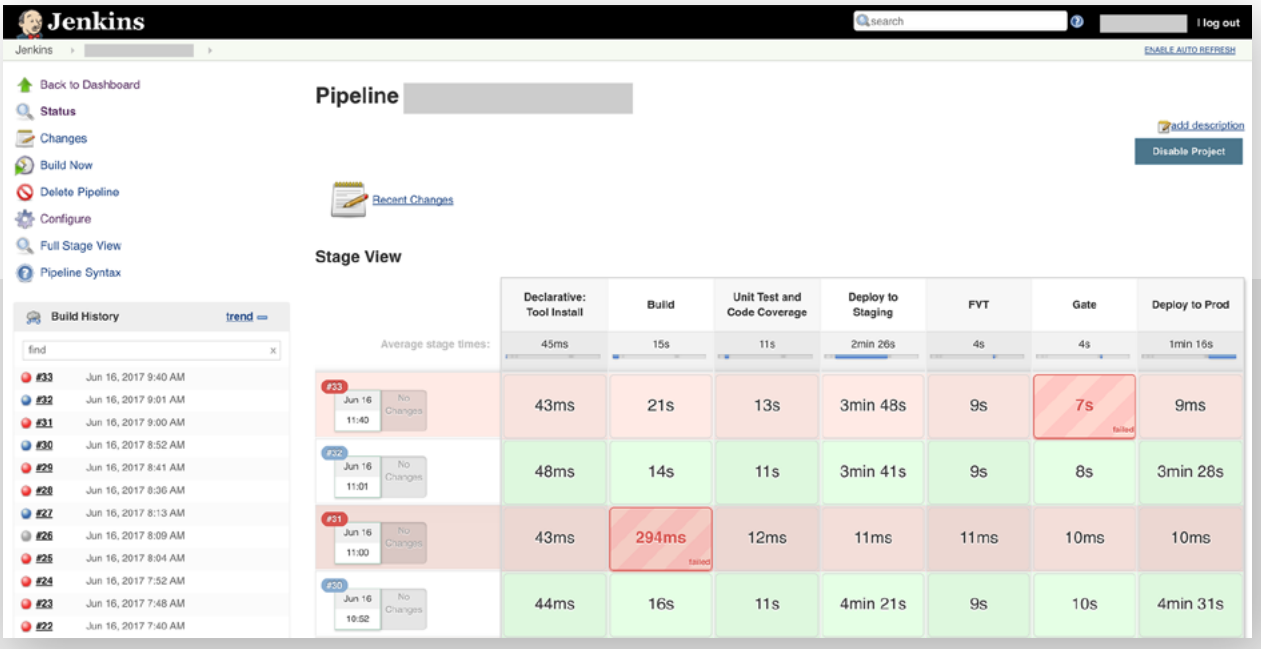
Harness leads the pack when it comes to governance, with fine-grained RBAC, full audit trails, and proprietary and integrated secrets management. It was built for cloud-native technologies but also works well with legacy tools.

© Harness Inc. 2021



Jenkins

Jenkins, a CI/CD solution that is now a decade old, wasn't made as a cloud-native solution; it was geared toward legacy technologies. It was initially built as a CI solution, but can be extended to handle CD as well. Jenkins, for all intents and purposes, is quite basic if you strip it down to its bones. Most capabilities and features come in the form of plugins, which traps you in dependency hell. Deployment automation? Scripts. RBAC? Plugin. Audit trails? Plugin. You get the pattern. While it does create pipelines that are super flexible and customizable, the amount of overhead in getting it set up and maintained (2-5 FTEs per year) is staggering, for a "free" solution.



Pros

Jenkins, through scripts, integrations, and plugins, can do it all. It truly is Frankenstein's CI/CD Monster, and is open-source.

- Open-source.
- Maintained frequently, transparent about security vulnerabilities.
- A decade old, so lots of community support—someone has most likely "been there, done that" in regards to issues.

Cons

Frankenstein's Monster, indeed. It is both a pro and a con that it can do everything, mostly because of the sheer amount of time, effort, and engineering costs it takes to set up, maintain, and operate Jenkins. It's a tough pill to swallow for users who chose Jenkins because it was "free."

- Dependency hell: plugins that create chains of dependencies.
- Not cloud-native, any advanced features require plugins or scripts.
- Requires tons of upkeep/engineering hours to maintain.

Pricing

Open-source—no pricing here. As we've been stressing, there are hidden costs to open-source software, and Jenkins is certainly a master of that arena. We've done our research, and as mentioned above, found that 2–5 FTEs are needed to maintain Jenkins—so while it may be "free," it truly isn't.

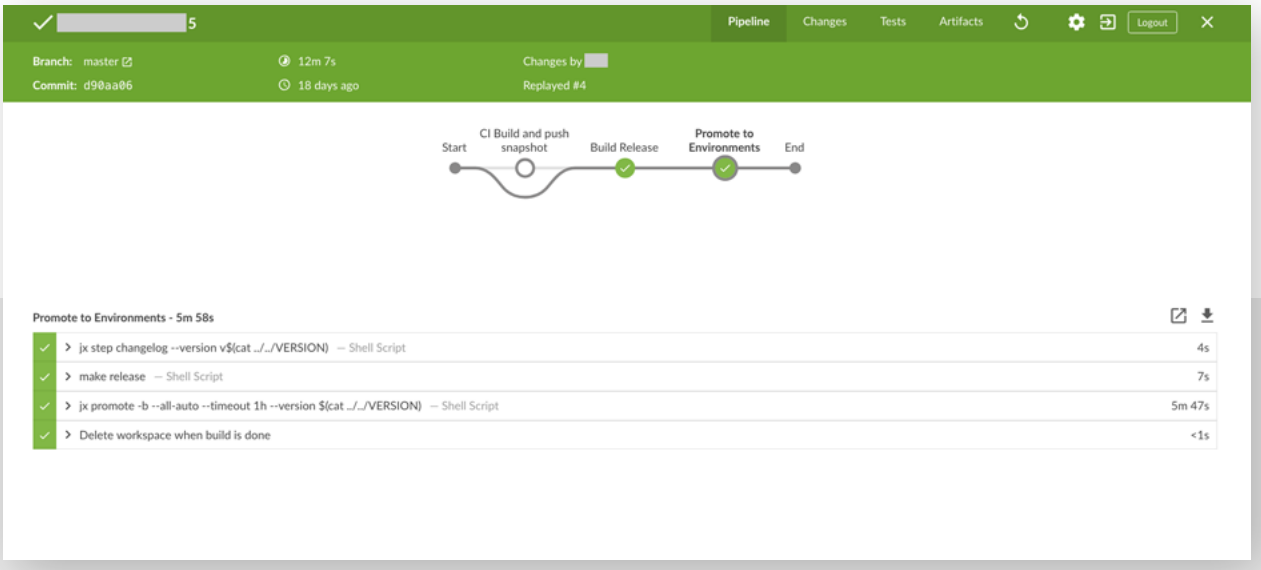
—

While it does create pipelines that are super flexible and customizable, the amount of overhead in getting it set up and maintained (2–5 FTEs per year) is staggering, for a "free" solution.

© Harness Inc. 2021

Jenkins X

Jenkins X is generally good at removing some of the complexity that plagues Jenkins. It is actually cloud-native, provides pipeline automation, and has built-in GitOps. While it is leagues more modern than Jenkins, it still comes with some of the same pain points and failures: no automatic dependency verification, no audit trails, it's only available on-prem, etc. While it is a marked improvement, we'd still advise organizations to find a tool that will be easier to maintain and has more modern features without requiring a ton of toil.



Pros

Jenkins X is cloud-native, unlike its predecessor, and is built on Kubernetes. It definitely dances circles around Jenkins when it comes to modern features—and UI.

- Simple UI, open-source.
- Preview environments, GitOps, cloud-native.
- Easy to scale, highly customizable.

Cons

Jenkins X still shares many of the same pain points as Jenkins. Compared to Jenkins, yes, it's leagues better—but compared to an end-to-end CI/CD tool like Harness, Codefresh, or GitLab, it simply cannot keep up.

- Migrating from another service to Jenkins X can be messy.
- Generally has the same failures/pain points than Jenkins.
- Doesn't support enough third party tools/integrations.

Pricing

Open-source—no pricing here. As we've been stressing, there are hidden costs to open-source software.

Jenkins X still shares many of the same pain points as Jenkins. Compared to Jenkins, yes, it's leagues better—but compared to an end-to-end CI/CD tool like Harness, Codefresh, or GitLab, it simply cannot keep up.

© Harness Inc. 2021

Author Appendix

Written By:

Roxanne Williams, Content Strategy Manager at Harness
Roxanne is a lover of all things DevOps, devoting a large portion of her time at Harness to competitor research. As such, she has developed a good understanding of where platforms shine—and where they can improve.

Reviewed By :

Bryan Feuling, Solutions Architect at Harness
Bryan began his tech career as a Help Desk Technician for a Fortune 200 company. His experience includes Database Administration, Application Development, Automation Engineering, and more. He has worked with hundreds of companies to help them avoid the same issues, pains, and engineer burnout that he saw and still sees in the industry.

Jim Hirschauer, Senior Director, Platform Marketing at Harness
Jim started his professional journey as an Aerospace Engineer, but his love for computers won. He held roles such as Systems Administrator, Monitoring Architect, and Solutions Engineer before jumping to the Evangelism side and finally, into Product Marketing here at Harness, where he manages a team and gets ship done.

© Harness Inc. 2021