8 REASONS FOR DEVOPS TO USE A BINARY REPOSITORY MANAGER

Publish: March 2024

INTRODUCTION

Enterprises are increasingly focused on software development operations, better known as **DevOps**, to identify and deploy the most effective solutions and best practices that increase the speed of deploying high quality releases, more frequently and securely into production environments.

To succeed in this mission, DevOps professionals must aim to increase efficiency and reduce friction at all points in the software development lifecycle (SDLC). This includes monitoring and automating workflows from curating open source packages during the coding phase through builds and CI/CD technologies to accelerate quality releases and secure distribution.

To this end, companies continue to invest significant resources in Digital Transformation. Unfortunately, to succeed in this endeavor, C-level executives and development managers need to make wise choices when it comes to selecting the right technology platforms for their organizations that will result in faster software delivery, higher quality, improved collaboration and ultimately drive business success in today's digital landscape.

$3.9 trillion

Global spending on Digital Transformation By 2027

51 percent

US technology executives who have not improved performance by investing in digital transformation

It’s also important to recognize that more and more of today’s businesses are migrating to cloud computing as the default model for applications, storage, and processing. The cloud enables organizations to innovate fast, and make bold development decisions at minimum cost and risk, while maintaining the ability to scale as required. In fact, migration to the Cloud is one of the best ways to improve software development efficiency today.

The Cloud is also playing a major role in the transformation of software development from the limited activity of a few small, tightly-knit groups to large numbers of developers coming from multiple interdisciplinary teams, oftentimes sprawled around the globe. For today’s enterprise, an iterative and incremental approach to software development has become the norm, based on environments that are cloud-enabled, software-driven, and virtually managed.

84 percent

Enterprises spending over $1.2m per year on public cloud

35 percent

Enterprises spending over $12m per year on public cloud

This White Paper, explores why a binary-centric approach to DevOps is the best way to succeed and presents the top 8 ways that JFrog Artifactory, the binary repository manager at the heart of the JFrog DevOps Platform, gives SREs and DevOps teams the tools to efficiently manage an ever-growing matrix of binaries, environments, and geographically distributed sites.

DEVOPS IS ALL ABOUT BINARIES

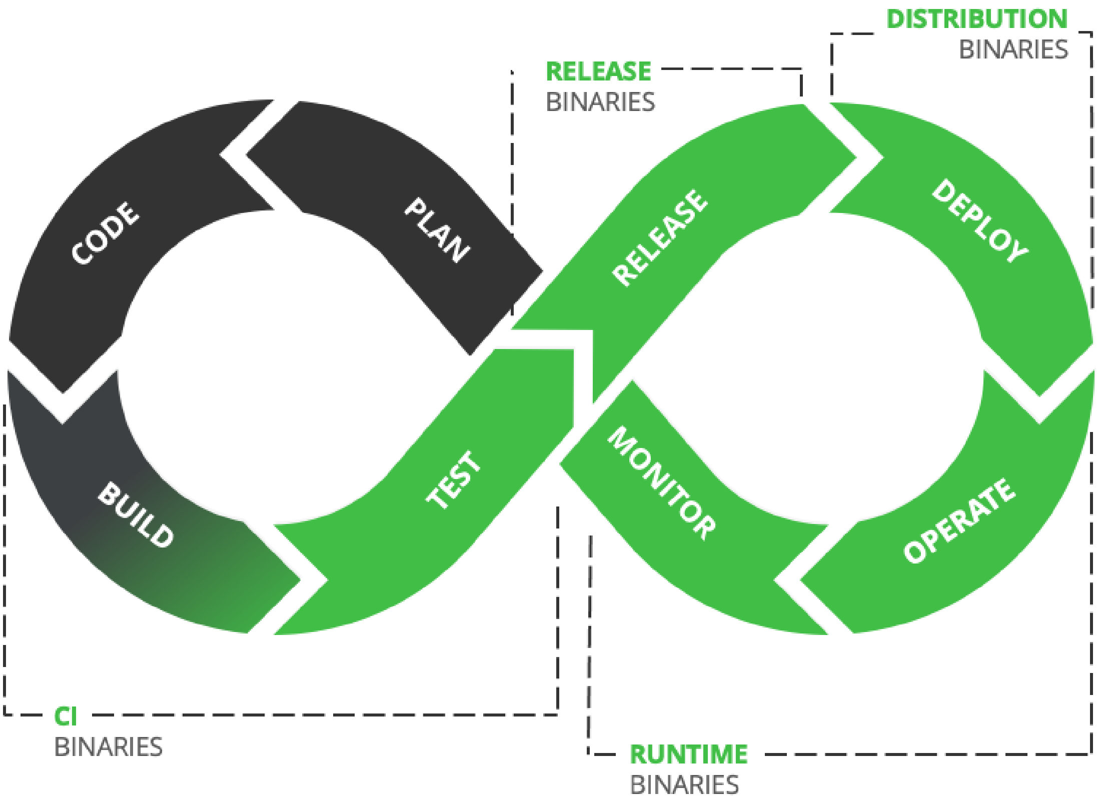
It’s perfectly legitimate to challenge this axiom. After all , how can DevOps be all about binaries, when software development is really all about code?" Of course it is, and developers need to be able to focus on writing innovative, bug-free source code that creates value for customers. To most developers, source code is what matters.

But as important as coding is, source code isn't what your customers use -- they run an application, which is a composite of many binaries from multiple sources. Therefore, the quality of software that is delivered to customers ultimately depends on the quality of the binaries that go into it.

Software development is about code, and quality code comes from smart people who know how to write it.

Software delivery is about binaries -- ensuring quality builds and getting them swiftly into your customers' hands. **Quality binaries come from smart systems that know how to manage and distribute them.**

We can see this in the standard diagram of the DevOps loop and SDLC best practices:



What's clear is that source code plays a foundational, yet proportionally small role in the total SDLC. Most processes in the DevOps loop don't deal directly with code at all, but instead center on the binaries that are built, tested, deployed, and run.

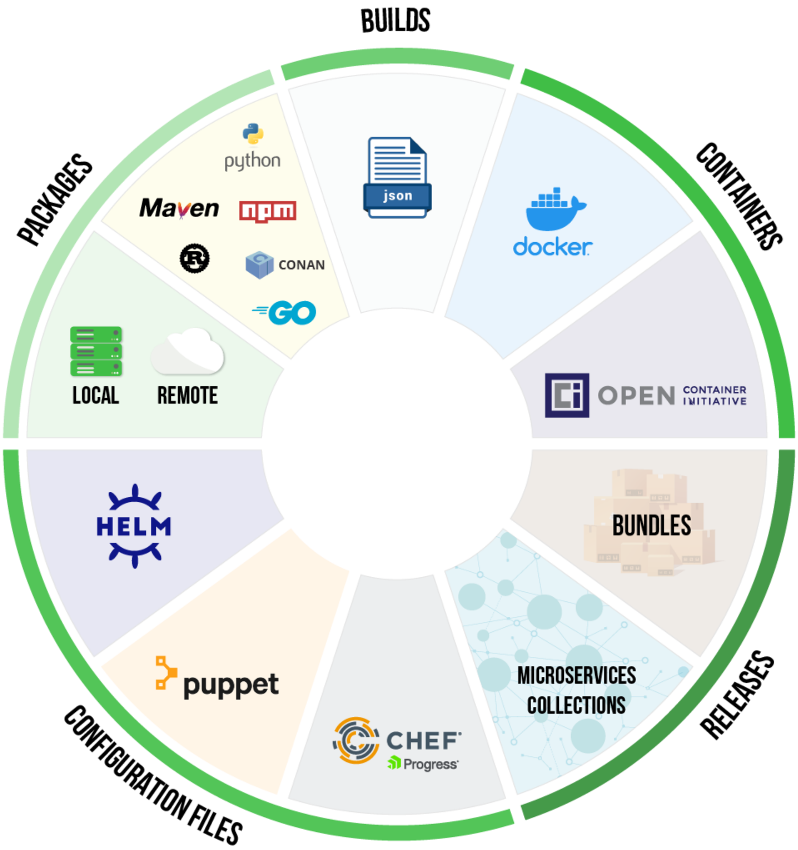
ARTIFACTS: WHERE DEVELOPMENT MEETS DELIVERY

Artifacts are the building blocks of software, from your packages and configuration files to the binaries that are the deployable runtime components of your application. Every delivered application is composed of a set of artifacts that connect to each other.

Your binaries are what is sent through your software delivery pipelines from end to end as they pass checkpoints of testing, validation, and security before final deployment to production.

While every application begins with source code, each source file both produces and relies on many artifacts, including binaries and dependencies, that go through many builds and test steps before resulting in a deliverable piece of software.

DevOps aims to **speed the delivery of quality software releases**. It does this by removing friction in the software delivery flow, which is most often found where software lifecycle processes meet. The consistency and certainty this brings enables automation for fast delivery.



Artifact Ecosystem

Your ecosystem of artifacts -- deployable binaries and their associated files -- occupy the space where development meets delivery. The JFrog Platform’s ability to **control, ease, and automate the flow of your binaries is the single most powerful key to successful development operations.**

FROM CLOUD NATIVE TO CLOUD NIMBLE

Systems architects must make sure their systems are cloud native, with applications that are optimized for scalable cloud technology infrastructure.

What's become clear is that cloud native solutions were only the start for running agile, scalable operations in cloud and hybrid environments. Forward-looking DevOps engineers must also seek solutions that are **cloud nimble**, meaning they support operationally consistent platforms that empower you to choose, and enable you to spread critical workloads across multiple domains both public and private.

Hybrid cloud and multi-cloud are fast becoming the most popular IT architectures for cloud service delivery, as enterprises seek even greater agility for their virtual operations through divestment from a single cloud provider.

Cloud nimble enterprises demand the ability to run anywhere, in any cloud infrastructure, , so they can:

- Match cloud choice to best-in-class services for given tasks
- Allocate workloads to the lowest-cost provider
- Shift workloads for load balancing oradjusting to demand pricing
- Keep data and compute services in geo proximity
- Maintain redundancy to help ensure five nines availability
- Employ a hybrid strategy to comply with security and/or regulatory requirements

A cloud nimble enterprise is better able to adapt to change without losing balance or speed.

Maintain a Consistent Security Posture
Protect your software supply chain with consistent permissions and monitoring across cloud domains.

Same Feature Set Everywhere
Interoperate across clouds with a consistent operating model through the same best DevOps practices.

Distribute Across Best-in-Class Services
Choose best-fit cloud providers for different segments of your SDLC.

Multi-Domain Automation
With reliable geo-replication and interoperability, securely deliver payloads from one cloud to another.

Transparent Cloud Migration
Maintain operational continuity migrating packages, builds, and metadata from one instance to another.

Choose Your Cloud For SaaS
Host SaaS accounts on multiple cloud providers for data proximity to the cloud ecosystems your teams use.

1 A SINGLE SOURCE OF TRUTH IN THE CLOUD

Adhering to a single set of workflow best practices becomes progressively harder as an organization adds languages and technologies to its toolset.

As the power behind the JFrog Platform, Artifactory is a universal repository manager for all of your artifacts, dependencies, binaries, and configuration files. With native support for **over 30 package types**, including generic repositories, it's your single source for storing, securing, and tracing the entire artifact ecosystem of your development organization. Developers can store and retrieve their packages or images in Artifactory repositories through the same package management services they use every day.

FRICTION POINT

Polyglot Development

Most enterprises use
10 or more
distinct package types

Almost
60 percent
of developers plan to adopt a new language In the coming year

Alpine	Cargo	Chef	CocoaPods	Conan	Conda	Cran	Debian	Docker
Gems	Generic	Git LFS	Go	Gradle	Helm	Ivy	Maven	Npm
NuGet	Opkg	Php Composer	Puppet	PyPI	Rpm	Sbt	Vagrant	Bower

Artifactory Repository Types

Artifactory's universal binary management serves all the developers in your organization. Whether they develop programs in Java, JavaScript, Python, Go, C++, or C#, Swift, or Rust,, Artifactory is the central home for everyone's packages and builds.

> *We haven't hit a kind of thing we can't push into Artifactory yet. Our technology choices aren't ever limited by Artifactory."*
>
> – Graham Bucknell, CI/CD Lead at Monster

When all of your binaries are governed through one common platform, your entire enterprise can align around the same SDLC workflows and best practices that will assure quality and accelerate release velocity. This is why Artifactory is the central component of a fully-automated software distribution pipeline that powers the JFrog DevOps Platform.

2 TRACK A SOFTWARE BILL OF MATERIALS (SBOM)

Making builds is easy. Knowing about everything you've built is hard. Especially when you make many new builds every day or every hour.

Artifactory stores new metadata -- what we call "build info" -- with every build you make, linking to the package metadata of your open source and proprietary dependencies along with build artifacts and environment settings. With detailed build info, you can **trace every build back to where it came from and out to every place it's been staged for service**.

Your build info is the basis of a Software Bill of Materials (SBOM) -- a machine-readable inventory detailing all the items included in an application and their origin -- for every release put into production or delivered to a customer.

FRICTION POINT

Artifact Counts

Over 5 million

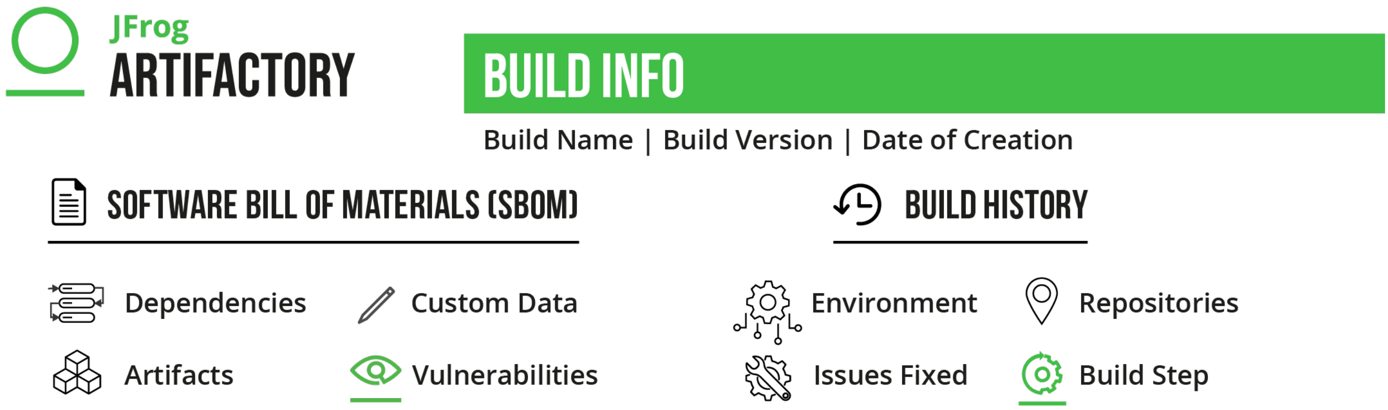
Average number of artifacts per enterprise

Nearly 50%

Enterprises that have an artifact size over 500 GB

As a growing number of governments and regulated industries require an SBOM to help combat cyberattacks, the JFrog Platform is your turnkey solution for compliance.

Artifactory's build info helps ensure that you are never in the dark about where a build came from, how it was created, or where it was deployed.



> *A big bonus for our developers is build metadata – when log4j hit, it was the easiest thing to generate a report of what apps had that vulnerable dependency, fix it, and we were good to go."*
>
> – Caio Trevisan, DevOps Service Owner, Bendigo and Adelaide Bank

ARTIFACTORY QUERY LANGUAGE

Artifactory Query Language (AQL) empowers DevOps engineers to quickly identify relevant packages and builds within growing repositories. They can employ AQL in their CI/CD servers to automate procedures that would otherwise require human intervention, or in analytics to improve workflow performance.

3 CACHE REMOTE REPOSITORIES BY PROXY

The open source software (OSS) dependencies drawn from remote resource repositories such as npm, Maven, Conan, and others can easily be the more significant portion of code in today's applications. Ensuring site reliability and speedy access is a vital key to maintaining release velocity, but can face several challenges:

Network latency
Inherent latency from physical distance - a remote site can be on the other side of the globe.

Heavy loads
Delays produced by heavy demand for a service.

Site downtime
The remote site where your dependencies are stored may suffer a service disruption from failure, attack, or loss of service.

Poor connectivity
Network outages, jitter, poor bandwidth, unstable connections.

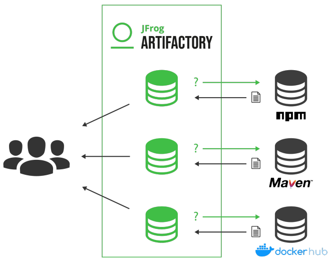
FRICTION POINT

Remote Repositories

Over
1.3 million
packages in the npm registry

96 percent
Enterprise applications that leverage open source code

A remote repository in Artifactory is a local proxy that caches your remote resource dependencies on the same infrastructure where you keep your local repositories. Developers never access a remote resource directly, but build using an on-demand copy of the dependency in Artifactory. When Artifactory is hosted on the same cloud environment as your compute, your builds can run at top speed.



> *Imagine a couple thousand developers going through a single proxy to be able to access, say, Maven Central. That unnecessary traffic is now negated because of Artifactory. We're able to build our solutions without going outside of our firewall."*
>
> - Mike Smith, Technology Architect at Kroger

Eliminating network latencies inherent in physical distance or across cloud service regions, helps keep builds running as fast as possible. The proxy also protects against disruption if the remote site itself is unavailable.

Equally important, the cache in Artifactory helps maintain your dependencies as immutable versions -- after the first on-demand pull from the remote resource, the package in the cache never changes. This guards against any force-push overwrite into a remote resource (possibly with malicious intent) and guarantees deterministic builds.

VIRTUAL REPOSITORIES

A virtual repository encapsulates any number of local and remote repositories, and represents them as a unified repository accessed from a single URL. It gives you a way to manage which repositories are accessed by developers since you have the freedom to mix, match and modify the actual repositories included within it.

4 BUILD ONCE AND PROMOTE EVERYWHERE

Under the agile methodology of continuous integration, every new software version must pass several quality gates in an SDLC. A candidate version is promoted successively to different teams for integration, testing, and staging before it is released for general use.

But what passes through these gates makes the difference between a speedy or a plodding path to release. When version promotion is done through source code, each receiving team must perform its own deterministic build of the code in their own runtime environment, potentially leading to a different set of binaries. This lacks certainty that the software evaluated at each stage is exactly the same as the last.

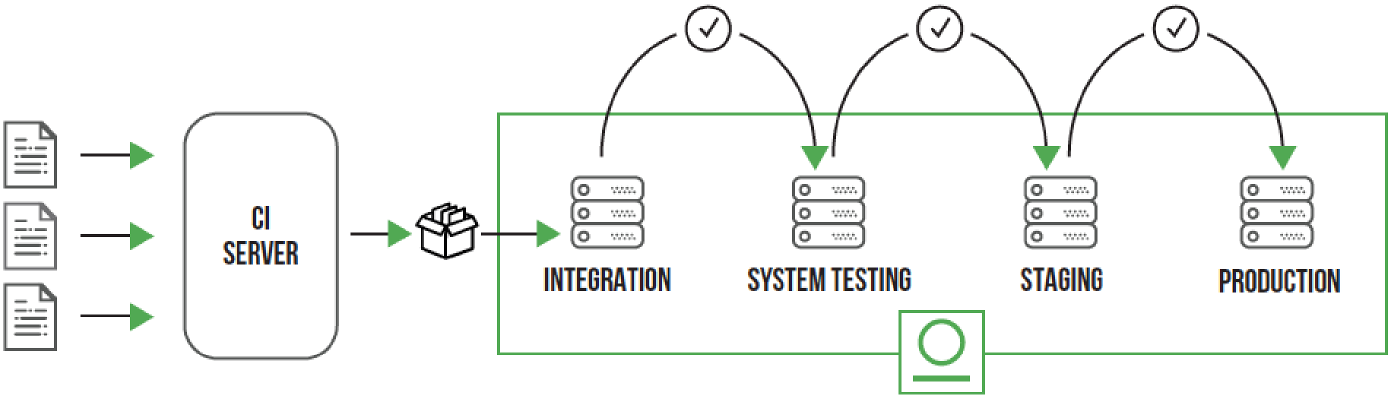
FRICTION POINT

Continuous Integration

3.8 hours

Average time developers spend coding per day

Artifactory's binary repository management offers a more consistent and reliable method that carries a single, immutable binary through the entire SDLC. With a repository for each SDLC stage, a build with its metadata can be promoted in the JFrog Platform simply by shifting it to the next repo in sequence.



In this "build once and promote" approach, the same build is evaluated at every stage, assuring absolute consistency through the DevOps pipeline.

Once free of having to perform their own builds or manage build environments, teams can apply the hours they recover to conducting more exhaustive tests and delivering feedback more quickly.

At each stage, teams can add additional metadata about stability, security, and more to the candidate's build info. By passing critical metadata learned from one stage that can be used by the next, the JFrog Platform accumulates **a comprehensive record about the lineage of every build put into production**.

CHECKSUM-BASED STORAGE

The JFrog Platform optimizes storage by ensuring that any binary is only stored once on the file system. When a binary is stored in a local or remote repository, Artifactory calculates a unique checksum (both MD5 and SHA1 are supported) of the file and renames the file to its checksum. Repositories hold only references to files and their metadata, so when a binary is copied or promoted to another repository only the references are changed. The physical file is never duplicated, and its checksum can be used to verify the binary's integrity.

5 ACCELERATE CLOUD NATIVE DEVELOPMENT

Increasingly, software development is cloud native development: applications written to effectively utilize cloud technology infrastructure, and enable the inherent best characteristics of running in the cloud.

This means producing container-based microservices, relying on established cloud native standards like OCI and cloud native tools like Docker and Kubernetes.

Docker repositories in the JFrog Platform fully support all Docker Registry APIs, so they can function natively with the Docker CLI. WIth local repositories, you can maintain as many private Docker registries in Artifactory as you need, to distribute and share container images within your organization.

FRICTION POINT

Cloud Modernization

80%
of edge applications are deployed to Kubernetes

Over $24 billion/year
spent on cloud application modernization globally by 2025

Artifactory reveals the layers within every Docker or OCI image that compose it, and links the metadata for a fully traceable path back to the origins of all its parts.

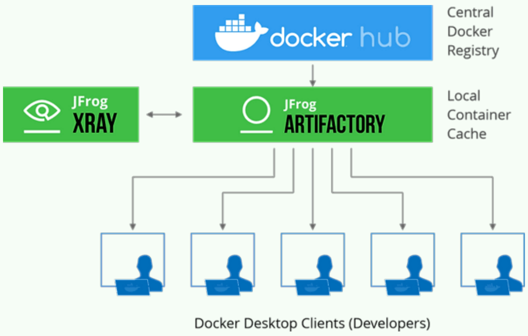
When combined with the JFrog Platform's fine-grained access control, you can maintain secure, private Docker repositories that exceed the security offered by Docker Trusted Registry. Using Artifactory's local repositories instead of private repositories on Docker Hub avoids all internet connectivity concerns, providing reliable and consistent access to images.

Artifactory also supports Helm chart repositories, so you can also manage your Kubernetes orchestration manifests alongside your Docker images. In this way, the JFrog Platform can serve as your comprehensive Kubernetes registry, a central, traceable home for everything deployed to your clusters.

NO LIMITS WITH DOCKER HUB

JFrog's partnership with Docker exempts JFrog Cloud users of Artifactory from Docker Hub's image pull rate limits. Ordinarily, anonymous free users are limited to 100 pulls per six hours, and authenticated free users to 200 pulls per six hours. But Docker Hub waives these limits when the image pull request is from a JFrog Cloud account.

By setting up an Artifactory remote repository to proxy Docker Hub, cloud users gain unlimited, high-performant access to Docker Hub and to Docker Official Images, simplifying cloud native application development.



By leveraging JFrog Xray, developers can also gain continuous, comprehensive vulnerabilities scanning of the images they pull from Docker Hub.

6 PROTECT YOUR SOFTWARE SUPPLY CHAIN

When managing enterprise development, protecting your software supply chain -- your application's ingredients -- from mistakes and attacks is critical to protecting what's most precious: your business.

Supply chain attacks like Tyupkin (2014), NotPetya (2017), Operation ShadowHammer (2019), and SolarWinds (2020) have spurred a heightened awareness and inspired the World Economic Forum to rank them among the top cybersecurity challenges.

As the single source of truth for all of your binaries, a binary repository manager makes an attractive target for a supply chain cyber attack. That's why **security is a top priority for the JFrog Platform**.

FRICTION POINT

CyberSecurity

2,200
Average number of cyber attacks per day

$10.5 trillion
Annual damage caused by global cyberattacks by 2025

Authentication

All services of the JFrog Platform require authentication through secure credentials such as a password or access token.

Checksum Verification

Every artifact's computed checksum is integral to how it is stored in Artifactory repositories, and is used to verify its integrity.

Permissions Management

With the JFrog Platform's fine-grained access control, administrators can ensure that developers and groups can access only the repositories through the CRUD operations they are authorized to.

SSO

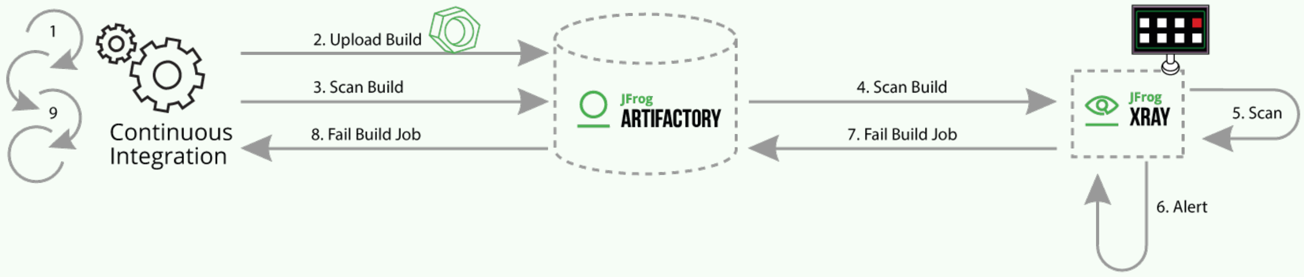
Support for LDAP, SAML, OAuth and SCIM protocols empower admins to integrate with Single Sign-On services such as Active Directory, Crowd and others for secure, organization-wide credentials management.

JFrog Artifactory's fine-grained permissions and its integration with single sign-on through multiple protocols empower organizations to maintain the same high-quality protection of their software supply chain across multiple cloud domains.

VULNERABILITY SCANNING

In a recent survey, only 18% of organizations reported being extremely confident in their open source components, which can typically be 60-80% of an application's code. Nearly twice as many were either not very or not at all confident.

The companion security solution JFrog Xray performs deep recursive scanning of binaries in your JFrog Platform repositories to identify all open source components that have known vulnerabilities. As part of the JFrog Platform, Xray is tightly integrated with Artifactory, maintaining additional security metadata and providing impact analysis so you can quickly remediate all binaries where a vulnerable dependency has been used.



With Xray, you can prevent high-risk builds from being deployed into production, and enable recall of builds with risks that are newly discovered.

In addition, Xray can also monitor the license types of your open source components, to alert you of all that are out of compliance with your organization's policies.

With JFrog Xray, you can operate with consistent monitoring, reporting, and remediation of vulnerabilities and compliance scanning across all cloud environments.

7 CONNECT TO YOUR UNIVERSE OF TOOLS

For developers, DevOps is a set of pipelines flowing in one direction toward a production-quality release.

But for the site reliability engineer, DevOps is an organizational fabric, weaving tools and procedures together to create a strong, resilient infrastructure.

Integrating a complex DevOps tool stack well is critical for reliable operations, and keeping this ecosystem running can consume far too much engineering time.

FRICTION POINT

Tool Stack Integration

Less than

50%

of SRE hours should be spent on operations

It starts with the tools that automates your builds: your CI/CD servers must integrate smoothly with the systems where they'll be stored. The JFrog Platform can connect to your choice for CI/CD automation through:

CLI
A command line interface (CLI) tool enables developers to store and retrieve binaries and metadata to and from repositories through a command window, shell script, or CI pipeline.

REST APIs
Developers can manage binary repositories using REST commands through curl or custom DevOps tools.

Webhooks
Trigger an action in another service in response to an event in Artifactory, either to notify users of the event or initiate an automated flow.

Build Integrations
Ready-made plugins and extensions for major CI tools such as Jenkins, Circle CI, TeamCity, Bitbucket, Pipelines, and Azure DevOps accelerate integration with your CI pipelines.

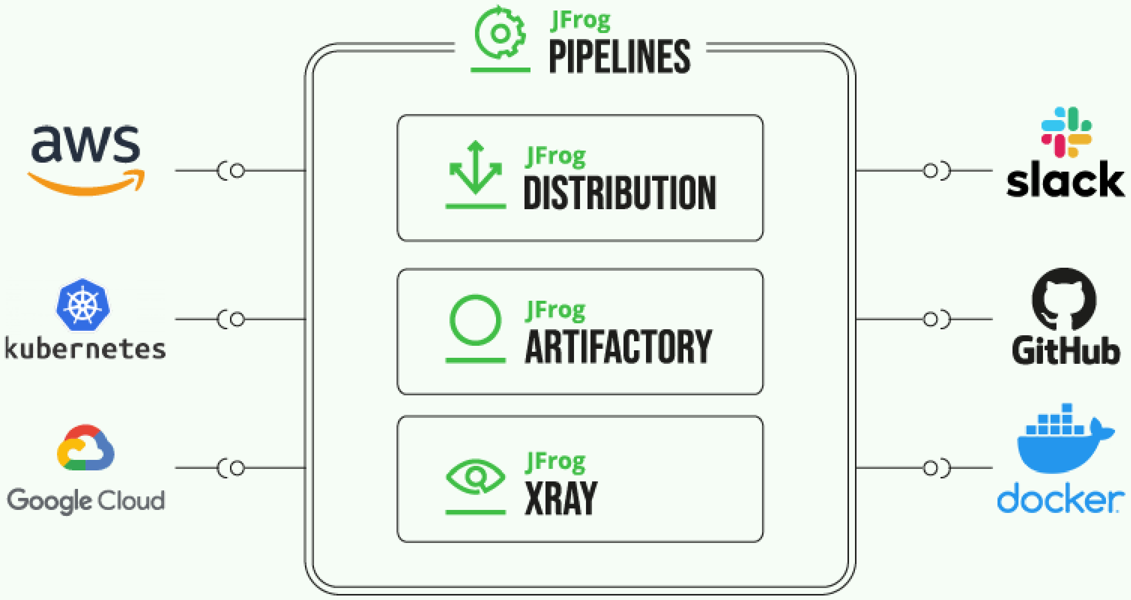
Of course, that's only the start of your integration needs. Tools for **automated testing, collaboration, ITSM, observability and analytics** are all part of your DevOps infrastructure fabric. You'll want to push the information these tools produce into your repositories, and connect that information to other tools, too.

With a large family of JFrog partner integrations with top industry providers, the JFrog Platform can help quickly build a strong, tight weave of your DevOps tool stack through this single source of truth.

NATURALLY INTEGRATED CI/CD

JFrog Pipelines CI/CD is the fastest way to build your CI/CD ecosystem. As part of the JFrog DevOps Platform, Pipelines is naturally joined with Artifactory. Using Pipelines' declarative pre-built steps, you can focus on what goes into your repositories, not how to get it there.

With the remaining components of the JFrog Platform, Xray, and Distribution, the unified tools at the heart of your DevOps universe are pre-integrated at installation.



Pipelines also provides out-of-the-box integrations for the most popular tools in your ecosystem, including GitHub, GitLab, BitBucket, Slack, Jira, AWS, GCP, Docker, Kubernetes, and many more. Integrating most of these services with Pipelines is as simple as entering a URL endpoint and user credentials.

8 SCALE TO INFINITY

For the modern enterprise, software development is now a highly collaborative endeavor of packages shared by intersecting teams across multiple sites and cloud regions spread across the globe.

Even small teams should expect to grow, and the best practices of DevOps have to stretch seamlessly along with them.

These binaries-centric best practices for DevOps - a single source of truth, metadata, build promotion, security, etc. - have been proven to scale smoothly. Every day, they enable JFrog Platform users to release quality software at speed whether it's developed by 5 people in one room, or 500 around the globe.

As your mission-critical tool, you'll need to ensure continuous and responsive access even as your department grows.

FRICTION POINT

Growth And Scalability

11 million

Average number of binaries per enterprise

28.7 million

Expected worldwide number of software developers by 2025

On-Premises High Availability
Self-manage as a BYOL installation in clusters in your own cloud account, or on servers in your own secure datacenter on-premises.

A High Availability configuration of replicated nodes in your cluster helps spread the load to accommodate large load bursts and ensures there is no single point of failure. This maximizes your uptime, even during system updates, up to "five nines" level of availability.

Cloud (SaaS)
Subscribe to a JFrog cloud managed service hosted in the major cloud provider of your choice (AWS, GCP, or Azure), and leverage the "any time, from anywhere" availability and elasticity of the cloud.

Employ a multi-cloud strategy by maintaining multiple SaaS accounts hosted on different cloud providers. This helps avoid vendor lock-in, and enables allocating workloads to the most cost-effective provider.

Hybrid
A hybrid strategy empowers you to employ the scalability of the cloud for your dynamic workloads, while keeping sensitive workloads in clusters within your secure on-premises data center.

The JFrog Platform's "same here, same there" promise of feature equivalency in every environment means you can divide your workloads between cloud and on-prem however you need.

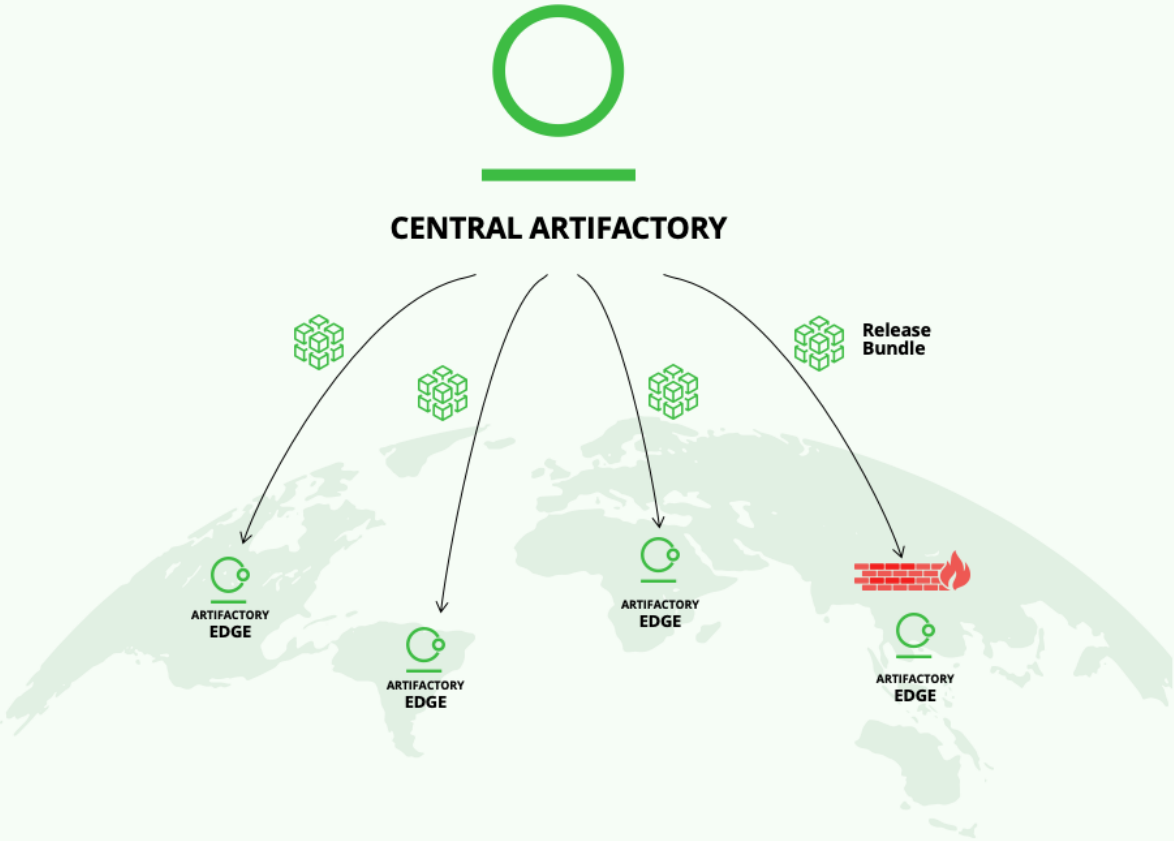
Multi-Site Geo-Replication
The JFrog Platform enables multi-site geo-replication through several push/pull replication topology options, or bidirectional federation of repositories.

JFrog's unique set of multi-site capabilities ensure locality in any global cloud topology. This empowers geographically distributed teams to work on the same artifacts (binaries and their metadata), with minimal latency so that every build, at every site, in every cloud region, can complete fast, without fail.

DISTRIBUTE TO THE FARTHEST EDGES

Once you have a fully validated release, where does it need to go? Enterprises must distribute software and updates to a growing number of global endpoints, for delivery to the clusters, devices, and desktops where it will benefit users.

As part of the JFrog DevOps Platform, JFrog Distribution empowers DevOps teams to efficiently package release bundles (binaries, artifacts, and metadata) from Artifactory and automate trusted software delivery to dozens or thousands of remote sites across the globe. Using JFrog Edge Nodes in multiple cloud regions -- cost-effective read-only Artifactory instances -- you can transmit signed, immutable releases through a private data network that are verified at each destination, while maintaining fine-grained access controls.



JFROG ARTIFACTORY PUTS YOUR BINARIES TO WORK

The evidence is clear: the key to successful DevOps -- minimizing or eliminating the friction in your pipelines -- lies in effective management of your binaries through your entire SDLC. A binary repository manager is your key agent of digital transformation.

The 8 reasons we've given here are the essential practices of this binaries-centered approach to DevOps. They're also the core values that the JFrog Platform is built around, to better assure your success.

Artifactory's design and features empower developers with the best practices of DevOps that reduce friction in key lifecycle stages, and accelerate software delivery.

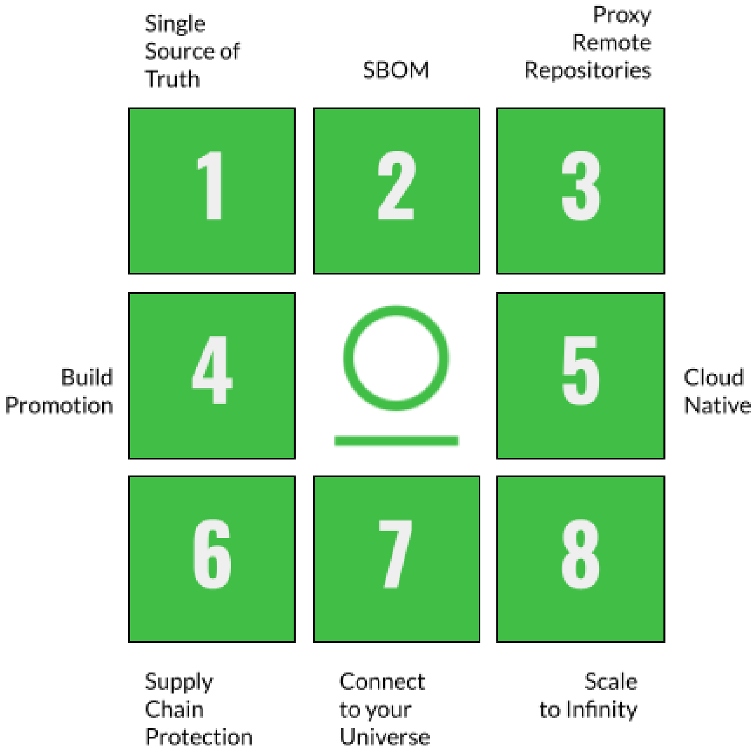
FRICTION POINT

Growth And Scale

Over 8300
organizations use JFrog

Over 70%
of the Fortune 500 use JFrog

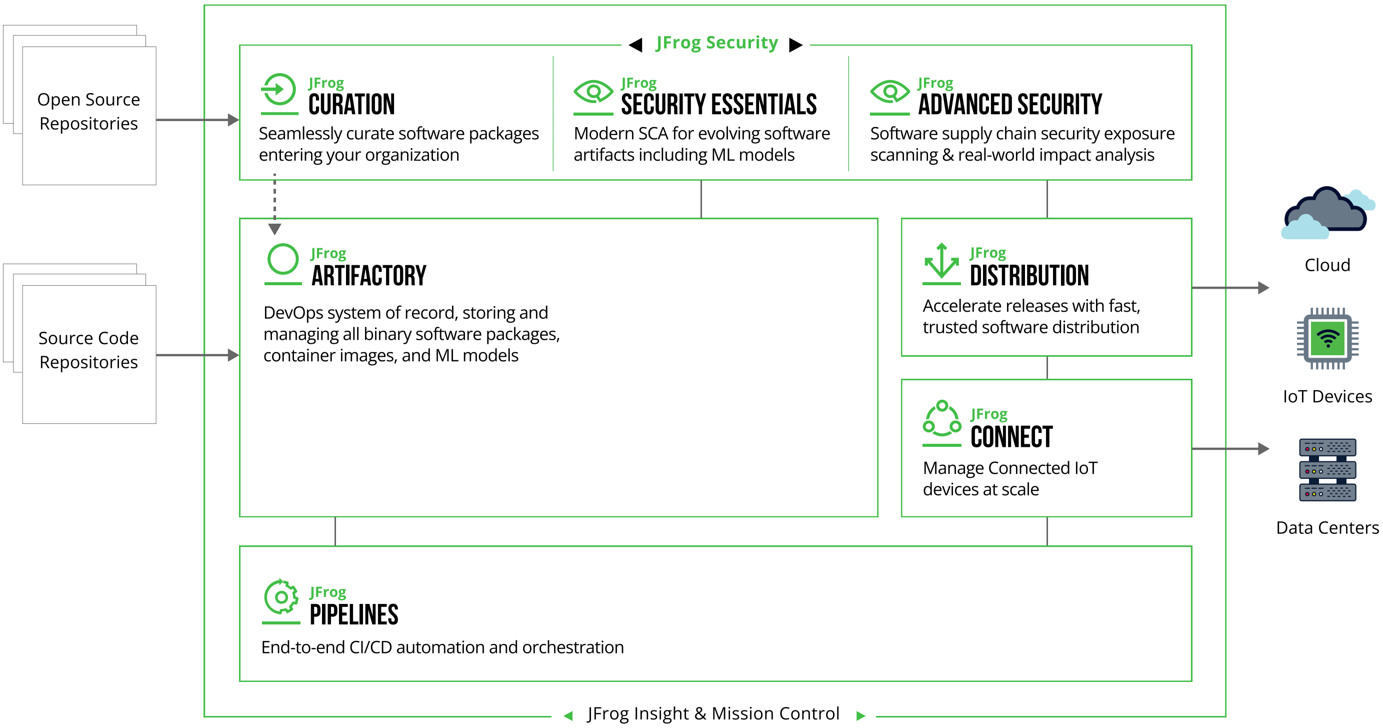
Over 42 billion
artifacts in JFrog Artifactory repositories



The effectiveness of this binary-centric approach to agile digital transformation is proven by thousands of JFrog customers every day, every hour.

JFROG PLATFORM

Artifactory is the core component that powers the JFrog DevOps Platform, a comprehensive end-to-end platform solution for one-stop DevOps.


Open Source Repositories
Source Code Repositories
◀ JFrog Security ▶
JFrog CURATION
Seamlessly curate software packages entering your organization
JFrog SECURITY ESSENTIALS
Modern SCA for evolving software artifacts including ML models
JFrog ADVANCED SECURITY
Software supply chain security exposure scanning & real-world impact analysis
JFrog ARTIFACTORY
DevOps system of record, storing and managing all binary software packages, container images, and ML models
JFrog DISTRIBUTION
Accelerate releases with fast, trusted software distribution
JFrog CONNECT
Manage Connected IoT devices at scale
JFrog PIPELINES
End-to-end CI/CD automation and orchestration
JFrog Insight & Mission Control
Cloud
IoT Devices
Data Centers

The cloud native JFrog Platform provides the full market-leading feature set in every operating environment, enabling you to operate your SDLC across clouds with the same best practices and consistent security posture for DevOps success.

The JFrog Platform is available at multiple subscription levels for self-hosted installations or as a managed service in the cloud. Each level includes a set of JFrog components appropriate to your organization's needs. Whether you're a small development team for internal applications or a global enterprise delivering secured software to the edge, JFrog has the right set of solutions for you.

SOURCES AND STATISTICS

- 84 percent of enterprises report public cloud spending of over $1.2 million , while 50 percent spend $6 million or more each year
  Flexera's 2024 State of the Cloud Report

- $3.9 trillion to be spent on digital transformation globally by 2027
  IDC Worldwide Digital Transformation Spending Guide

- 51% of US technology executives have not improved performance from their investment in digitala transformation
  KPMG's 2023 US Technology Survey

- Most enterprise software development operations use 10 or more istinct package types
  JFrog internal data

- Almost 60 percent of developers expect to adopt a new language in the next 12 months
  JetBrains State of Developer Ecosystem 2023

- Over 5 million artifacts per organization
  JFrog internal data

- Nearly 50% of enterprises have an artifact size of 500 GB or more
  JFrog internal data

- Over 1.3 million packages available in main npm registry
  npm blog, 2020

- 96 percent of current applications use open source code
  The Internet was Built on Open Source

- The average developer spends only 3.8 hours coding each day
  State of Developer Happiness Report

- 80% of edge applications are deployed to Kubernetes

- Over $24 billion/year spent on cloud application modernization globally by 2025
  Research and Markets, Application Modernization Services Market...Global Forecast to 2025

- According to Security Magazine, there are over 2,200 cyber attacks every day
  Top Cybersecurity Statistics for 2024

- Damage from global cybercrime expected to reach $10.5 trillion annually by 2025
  2022 Official Cybercrime Report

- Less than 50% of SRE hours should be spent on operations
  Google Site Reliability Engineering

- Average of 5 servers with Artifactory per enterprise
  JFrog internal data

- Over 11 million binaries per enterprise account
  JFrog internal data

- The number of software developers is expected to reach 28.7 million by the end of 2024
  Software Development Statistics