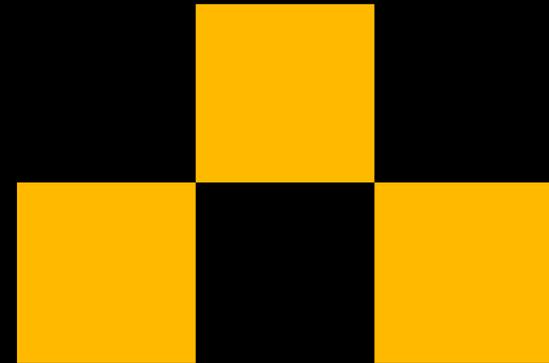


8 REASONS

FOR DEVOPS TO USE
A BINARY REPOSITORY
MANAGER

WHITE PAPER



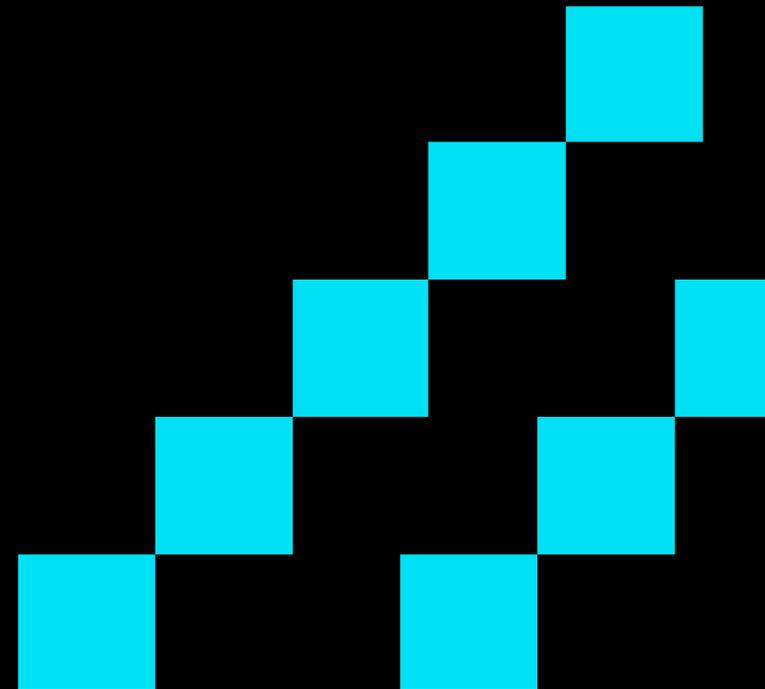
INTRODUCTION

Over the last several years, software development has evolved from deploying products periodically to building them on an ongoing basis using CI servers. A company's end product may be built on a daily or even hourly basis. This means that DevOps must support the continual flow of code from the individual developer's machine to the organization's production environment. Bridging the gap between the development teams' desktops and the organization's servers, load balancers and databases hosted on production systems can present several challenges.

Some of these are:

- » Providing stable and reliable access to repositories
- » Supporting a large number of common binaries across different environments
- » Security and access control
- » Tracing any action done to a file back to the user
- » Transferring a large number of binaries to a remote location
- » Managing infrastructure configuration across different environments

This white paper describes how a Binary Repository Manager such as Artifactory gives a DevOps team the tools to efficiently manage an ever-growing matrix of binaries, environments and geographically distributed sites to ensure a smooth application development workflow.

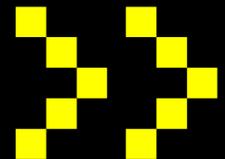
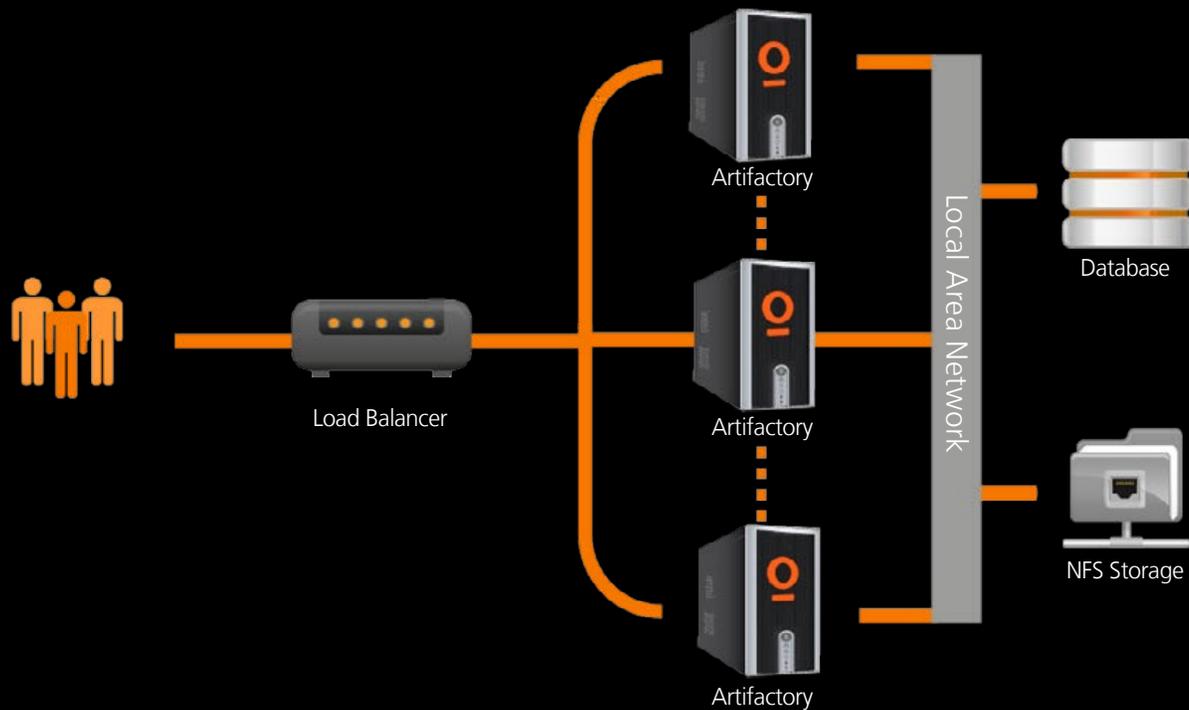


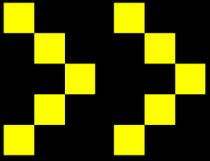
1 SYSTEM STABILITY AND RELIABILITY WITH ARTIFACTORY HIGH AVAILABILITY (HA)

Playing such a central role in development of your products, the servers hosting your binaries (whether you downloaded them or developed them) can become mission-critical components of your organization. As DevOps, it's your responsibility to keep those systems running, and any downtime

can have severe consequences for your development workflow.

Artifactory supports a **High Availability** configuration with a cluster of 2 or more active/active Artifactory servers on the same Local Area Network.





This redundant network architecture has several benefits:

No single-point-of-failure

Your system can continue to operate as long as at least one of the Artifactory nodes is operational. This maximizes your uptime and can take it to levels of up to “five nines” availability.

Accommodate larger load bursts

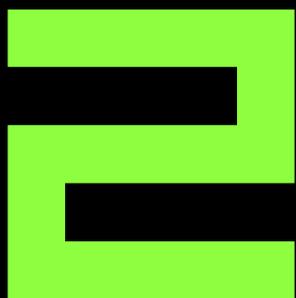
With horizontal server scalability, you can increase your capacity with no compromise to performance and meet any increasing load requirements as your organization grows.

Multiple server architecture

Artifactory HA lets you perform most maintenance tasks with no system downtime.

HIGH AVAILABILITY SYSTEMS

Systems that are considered mission-critical to an organization can be deployed in a High Availability configuration to increase stability and reliability. This is done by replicating nodes in the system and deploying them as a redundant cluster to remove the complete reliability on any single node. In a High Availability configuration there is no single-point-of-failure. If any specific node goes down, the system continues to operate seamlessly and transparently to its users through the remaining, redundant nodes with no down time or degradation of performance of the system as a whole. [Learn more >>](#)

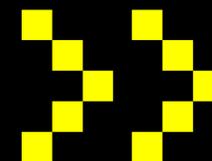


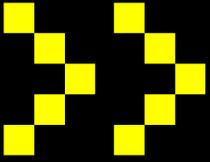
MANAGING MANY BINARIES ACROSS DIFFERENT ENVIRONMENTS

As the number of binaries used by your organization grows (whether built in-house or downloaded), and the number of development teams grows, managing those binaries across development, QA, production and possibly more environments needed by each team can become a maintenance nightmare. Creating the required separation by keeping different copies of the same file across different partitions of your file system might work, but this needs to be strictly controlled, requires intensive management, and can quickly become... unmanageable.

Artifactory removes the need to create different copies of the same file. Only a single copy of any binary is ever stored on the file

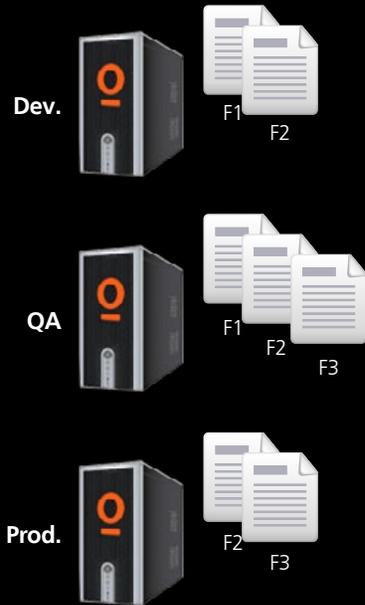
system, and Artifactory manages access to the file through repositories. Repositories behave like virtual file systems which you can dedicate to development groups and their respective environments. Access by the different teams (and the build server) is controlled by defining corresponding repositories and configuring access control on the binary as it goes through the development lifecycle. If you now consider the matrix of binaries, development groups and environments in your system you can understand the benefit of managing your binaries through repositories in a Binary Repository Manager such as Artifactory. No need to copy files or replicate environments to support the development cycle.





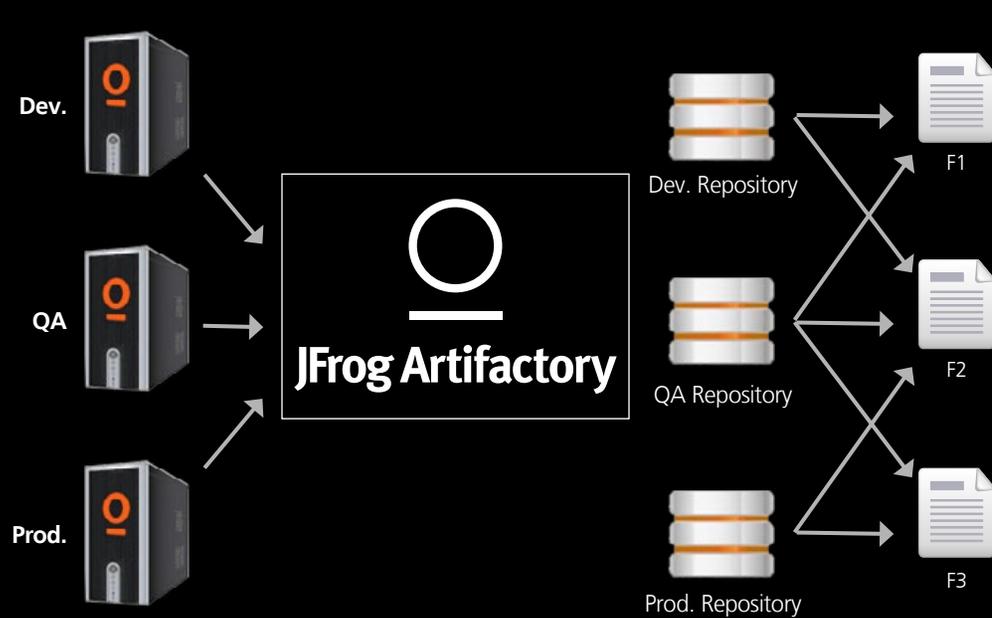
Before

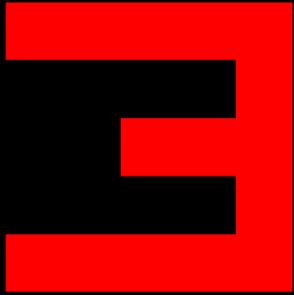
Each group maintains a copy of all files that they need.



After

Only one copy of each file is stored. Access is managed using dedicated repositories through Artifactory.





SECURITY, ACCESS CONTROL AND TRACEABILITY

Every organization needs to implement security policies so that people can only access internal resources that they are authorized to use. But how do you control what people in your organization download from external resources? How do you control which external resources are accessed in the first place? And then, how do you control where people in the organization put different binaries they downloaded or are working on? As the number of binaries that you manage increases, controlling access to them through the development cycle becomes more and more challenging.

Artifactory can provide security and access control at several levels. From restricting complete repositories down to restricting a single binary, and from a group of any size down to a single developer.

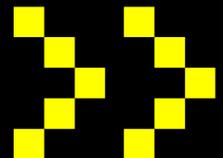
Virtual repositories let you restrict developer access only to reliable 3rd party resources that have been approved.

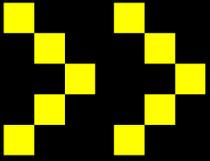
“Excludes” and “Includes” let you use naming patterns to define anything from a whole repository that should be excluded from your organization’s access, to including a single artifact within a repository which may be critical for your development efforts.

Permission targets and permissions can be defined to configure user or group access privileges to different resources that have been downloaded.

Integration with LDAP, SAML, Crowd and others manages access to your servers.

Comprehensive log information lets you trace virtually any action done on any artifact managed by Artifactory.





These comprehensive security and access control capabilities help you manage your development process by ensuring that developers can only access repositories for which they are authorized to.

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. You can also optimize artifact resolution by defining the underlying repository order so that Artifactory will first look through local repositories, then remote repository caches, and only then Artifactory will go through the network and request the artifact directly from the remote resource. For the developer it's simple. Just request the package, and Artifactory will safely and optimally access it according to your organization's policies. [Learn more >>](#)

4

FULL SUPPORT FOR DOCKER

As Docker technology continues to evolve, its usage continues to grow. If you are not yet using Docker in your organization, it is likely you will do so soon. So now, in addition to all the other package types you are managing, you also need to manage Docker images. But there's no need to onboard and maintain another tool. Artifactory is a fully-fledged Docker repository supporting all Docker Registry APIs. This allows the Docker client to work with Artifactory directly, presenting several benefits for enterprise Docker users.

Using local repositories, you can **distribute and share images** within your organization to make managing images between different teams easy. You can even replicate your Artifactory Docker repositories to remote instances of Artifactory to share images with colleagues in geographically distant sites.

Artifactory offers **fine-grained access** control to your organization's images with secure "docker push" and "docker pull" effectively providing **secure, private Docker repositories** that exceed the security offered by Docker Trusted Registry.

Using Artifactory, instead of private repositories on Docker Hub, removes any issues related to internet connectivity resulting in **reliable and consistent access to images**. And with Artifactory running in a **High Availability configuration** you get system stability and availability of your Docker images that is unmatched in the industry.

Artifactory's **smart search** makes it easy to find any Docker image stored in your system. Full support for the Docker Registry API supports basic search with the Docker client, but Artifactory offers much more. Built in searches answer common needs with single-click operations, custom properties provide the flexibility to meet a variety of specific needs, and Artifactory Query Language offers a simple way to formulate complex queries letting you find images based on any set of criteria.

Whether you're already on board with Docker or just evaluating how to introduce it to your organization, once you're using Artifactory to manage your packages, you're already covered for Docker images.

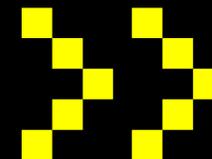
S

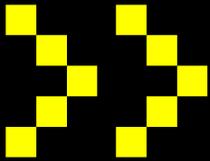
QUICKLY REPLICATE REPOSITORIES TO REMOTE SITES

Using traditional file-system storage, replicating a repository means that if the same file is stored in several different places, it needs to be synched over to the remote site several times. As the number of binaries you need to manage grows to accommodate different stages in the build automation flow (Dev, QA, Production) and different customers (both internal and external), the amount of data you need to replicate to remote sites can multiply very quickly, and can consume huge amounts of network resources taking a considerable amount of time.

Artifactory supports replication of your repositories to another instance of Artifactory which is outside of your local network. Replicated repositories are automatically synchronized with their source periodically, so that your artifacts can be made available to different teams wherever they may be located around the world. Artifactory's **checksum-based storage** optimizes the

replication process making it up to 10 times faster. Since any artifact is stored within a repository only once, and “copies” are actually metadata references to the same physical file, the amount of data that needs to be copied to the remote site when replicating a repository is dramatically reduced. The actual binary file only needs to be copied over once. Any additional “copies” represented by metadata, are much smaller than the file itself. As an added benefit, the metadata contains extensive log information regarding the physical file to which it refers. This means that any file within the replicated repository is copied over with full traceability. All relevant log information is stored in the metadata which is actually copied over.





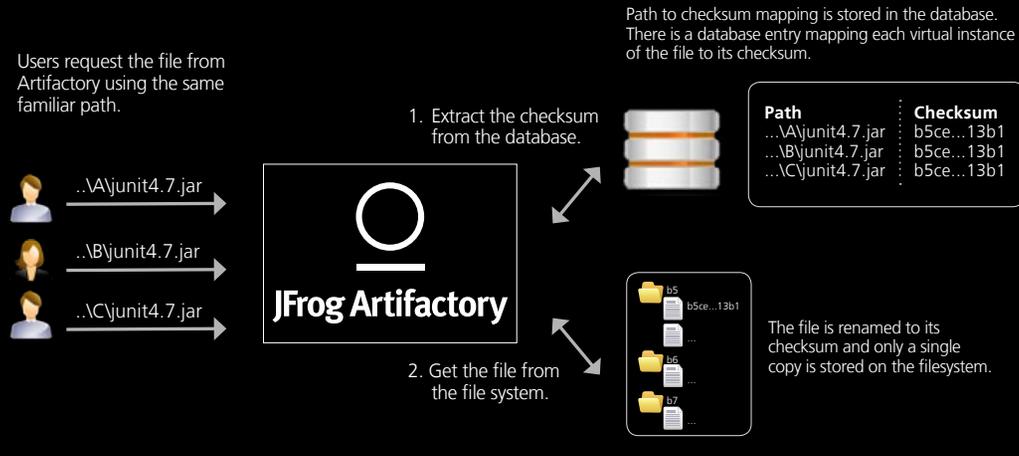
Regular Storage

Each copy of a file is stored in a different location in the file system. Each user accesses it using the path he is aware of.



Checksum Based Storage

Users request the file from Artifactory using the same familiar path.



CHECKSUM-BASED STORAGE

Artifactory optimizes storage by ensuring that any binary is only stored once on the file system. When a binary is uploaded to a local repository, or downloaded to a remote repository, rather than storing the file in its original name under a specific path, Artifactory creates a checksum (both MD5 and SHA1 are supported) of the file and renames the file to its checksum. All the metadata about a file is then stored in the Artifactory database. Therefore, to create a copy of a file in a different location, there is no need to actually copy the file, only the metadata is copied and this maintains a reference to the same physical file. [Learn more >>](#)

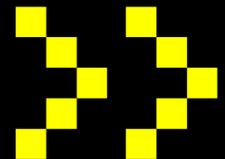


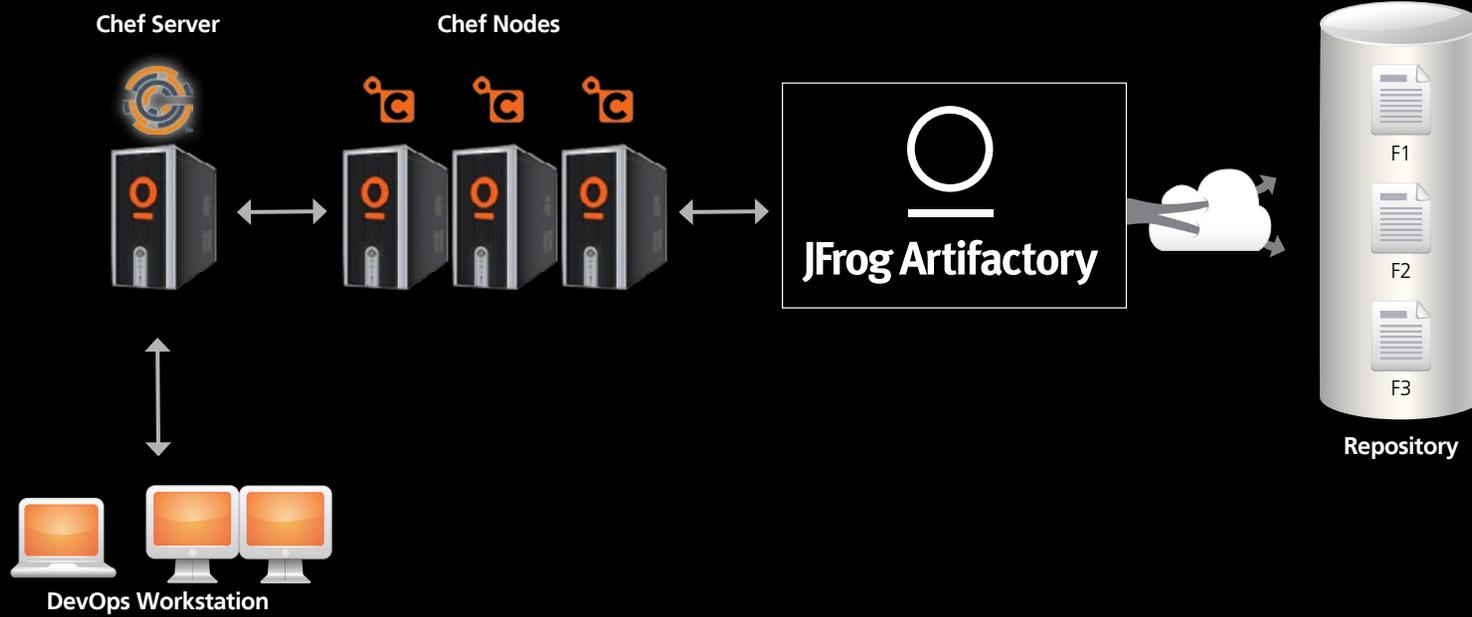
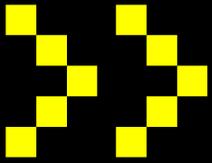
FULL SUPPORT FOR CHEF TO MANAGE INFRASTRUCTURE

Part of your role as DevOps is to manage different infrastructure configurations depending on the environment - development, QA or production. Each environment may have different yet overlapping requirements for setup, and you need to be able to manage all the packages that should be installed easily and efficiently.

Artifactory can make configuration a quick and painless process. You can define specific repositories per environment and direct your CHEF configuration script to the right repository in Artifactory. This ensures that each server you

configure will only get the packages approved for the designated environment. And if there are overlapping requirements between the different environments, there is no need to maintain multiple copies of any package. Artifactory manages that for you with internal references.





ADVANCED PACKAGE MANAGEMENT WITH "WATCHES" AND USER PLUGINS

While organizations may share common best practices in how they manage their packages, each organization is different and may have some unique requirements.

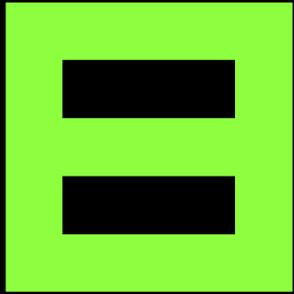
In Artifactory you can define "watches" which let you closely monitor any of the critical packages in your system. In case a package is accidentally deleted or moved, you can receive an immediate alert to restore it as required. But Artifactory also provides much more flexible and advanced package management capabilities.

Artifactory generates an event for virtually every action performed on a package including download, deployment, moving, copying and more. For every such event you may write a **user plugin** that effectively extends the Artifactory REST API giving you a simple way to add functionality to Artifactory, and implement complex, custom behavior in your binary management workflow. This includes scheduling tasks, managing security and authentication, license compliance and open-source governance, artifact resolution and deployment, build integration and promotion logic, maintenance and cleanup and more.

USER PLUGINS

User Plugins give you a way to extend the functionality of Artifactory with any custom behavior required by your organization. To keep things simple, they are written as Groovy scripts and have a simple DSL to wrap them as closures within the extension points. The plugins can be changed and redeployed on-the-fly, and can even be debugged - all from within your favorite IDE.

[Learn more >>](#)



A UNIVERSAL, END-TO-END SOLUTION FOR ALL BINARIES

No single packaging format or technology is sufficient to support development in a modern organization. There is a multitude of formats, a variety of build tools, different continuous integration systems and other technologies that go into building a flexible and maintainable software development ecosystem. Managing binaries for all the different packaging formats and integrating with all the moving parts of the ecosystem can become a maintenance nightmare.

Artifactory was designed from the ground up to fit in with any development ecosystem. Uniquely built on checksum-based storage, Artifactory supports any repository layout and can, therefore, provide native-level support for any packaging format. Essentially, regardless of the packaging format you are using, Artifactory can store and manage your binaries, and is transparent to the corresponding packaging client. The client works with Artifactory in exactly the same way it would work with its native repository. For example, if you are working with Docker, Artifactory proxies Docker Hub (or any other public Docker registry), lets you store and manage your own images in local Docker repositories, and works transparently with the Docker client. If you are working with node.js, Artifactory proxies npmjs.org (or any other public Npm repository), lets you store your own packages in local Npm repositories, and works transparently with the Npm client. Similarly for Vagrant, NuGet, Ruby, Debian, YUM, Bower, Python and more.

But development is only one end of the software delivery pipeline. Before a package makes it into a product, it needs to go through processes of build and integration. There are many build and integration

tools on the market, but there is only one product that works with them all. Through a set of plugins, Artifactory provides tight integration with popular CI systems available today such as Jenkins, Bamboo and TeamCity. These systems use Artifactory to supply artifacts and resolve dependencies when creating a build, and also as a target to deploy build output. And to support cloud-based CI systems on which you are not able to apply plugins, Artifactory provides plugins for the build tools you use (such as Maven and Gradle) which ultimately provides the same level of build automation. That takes care of development and deployment, but what about distributing your software once it's ready for consumption. That's where Bintray comes in.

Bintray is JFrog's download center in the cloud offering rapid downloads, fine-grained access control, detailed stats and logs and an extensive REST API. Promoting releases for distribution from Artifactory is a matter of a single-click or API call. Like Artifactory, Bintray is package-agnostic and works seamlessly with all the different package clients, so it can be fully integrated into any continuous integration/continuous delivery ecosystem.

Artifactory is a universal repository. It is the single tool that sits in the center of your development ecosystem and "talks" to all the different technologies, increasing productivity, reducing maintenance efforts and promoting automated integration between the different parts. Together, Artifactory and Bintray are the central components of a fully-automated software distribution pipeline.

SUMMARY

This white paper has shown how DevOps can use a Binary Repository Manager such as Artifactory to support the development cycle in a continuous integration environment.

From quick and easy configuration of servers to secure and reliable access to repositories, Artifactory provides DevOps with a tool to efficiently manage binaries across the different environments of your organization.

For more information on how Artifactory can boost your organization's performance please contact us at info@jfrog.com.