DEVOPS 使用二进制制品仓库管理工具的 8 大理由

引言

现如今所有公司都是软件公司；软件已占领全世界。无论是在技术、金融、制造、医疗保健还是娱乐领域，应用程序现在是所有公司的核心价值驱动力。

开发软件已经从密切配合的小型团队活动转变为由许多跨学科团队执行的活动，这些团队往往遍布全球。目前，迭代与增量式软件开发是大多数企业的常规解决方案。

曾经每年发布一两次重要新版本的组织，现在普遍每月或每周都向客户发布新版本。现如今，许多企业**每天多次**将更新部署到生产环境中。

随着发布越来越频繁，终端用户的期望也越来越高——客户要求频繁更新软件，以修复错误、修补漏洞、提供新功能。

截至 2023 年全球用于数字化转型的总支出 6.8 万亿美元

尝试数字化转型但未能实现目标占比 70%

这就是企业越来越关注 DevOps 的原因，DevOps 是更频繁地创建、部署更多高质量版本，并将其部署到生产环境中所使用的工具和最佳实践。为了取得成功，DevOps 实践旨在减少软件开发生命周期 (SDLC) 中所有的摩擦点，并实现工作流程的自动化（通过 CI 和 CD 服务器等），从而大大加快发布速度。

我们之后会解释为什么以二进制制品为中心的 DevOps 方法是获得成功的最佳途径，并且将展示位于 JFrog DevOps 平台核心的二进制制品仓库管理工具 Artifactory 为 SRE 和 DevOps 团队提供工具的 8 种主要方式，以高效管理不断增长的二进制文件，以及环境和地理上分散的站点。

与 JFrog Platform 中的支持解决方案一起，Artifactory 可确保流畅、安全、端到端的应用开发工作流程。

DevOps 的成功取决于二进制文件

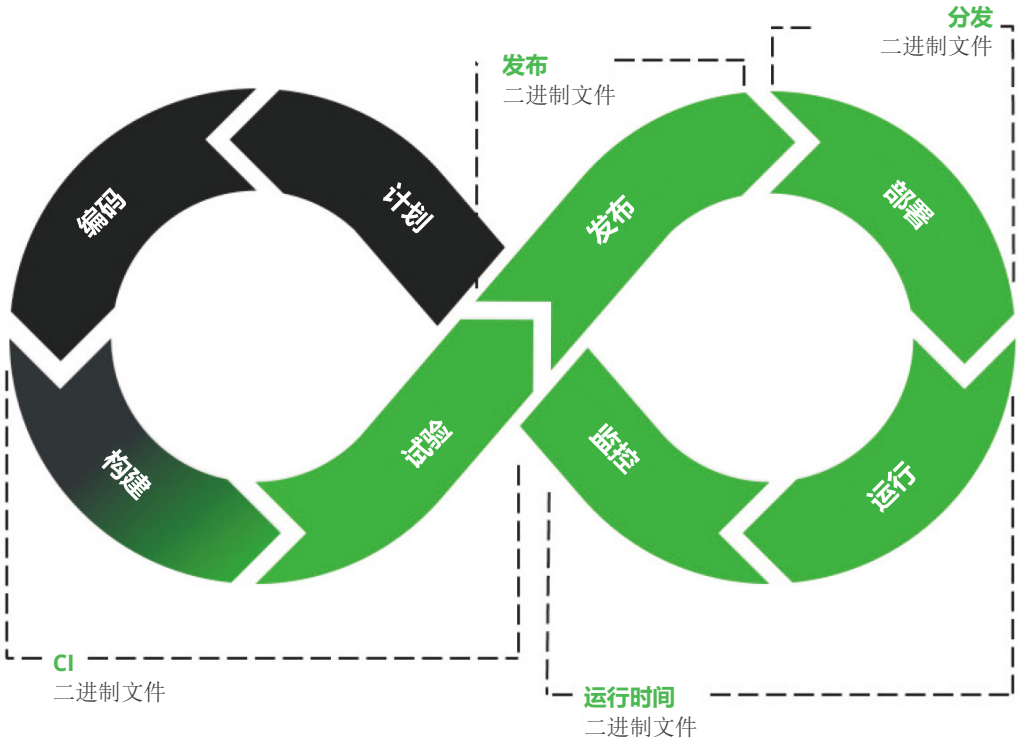
您可能会问，“DevOps 怎么会跟二进制文件有关呢？软件开发不都是写代码吗？”确实如此，开发人员需要专注于编写创新、无漏洞的源代码，为客户创造价值。对于大多数开发人员而言，源代码至关重要。

但与编写代码同样重要的是，客户所使用的并不是源代码——他们运行的是应用程序，是来源众多的二进制文件的组合。交付给客户的软件质量取决于其中包含的二进制文件的质量。

软件开发与代码有关，高质量的代码来自那些了解如何编写代码的有才之士。

软件交付与二进制文件有关——确保高质量的构建，并迅速交付到客户手中。**高质量的二进制文件来自那些了解如何管理和分发这些文件的智能系统。**

我们可以在软件开发生命周期最佳实践的 DevOps 循环标准图中看到这一点：



什么是显而易见的？源代码在整个软件开发生命周期中起着基础性作用，但所占比例很小。DevOps 循环中的大多数过程根本不直接处理代码，而是以构建、测试、部署和运行的二进制文件为中心。

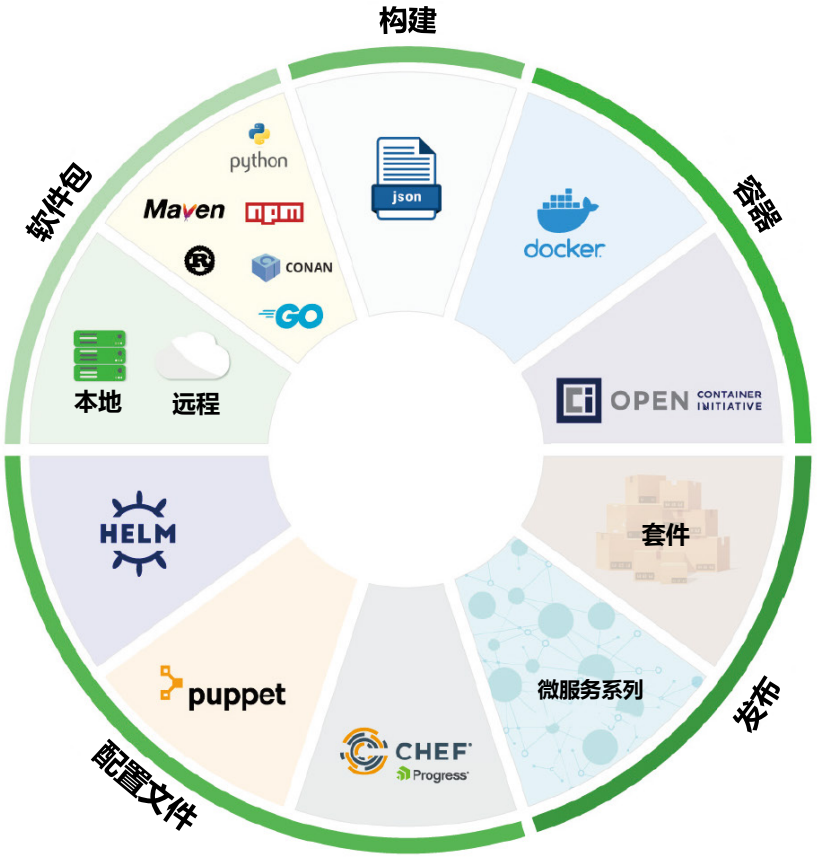
制品：开发与交付的中间产物

制品是软件的构建块，既是软件包、配置文件，又是应用程序可部署的运行时组件的二进制文件。交付的每个应用程序都由一组相互连接的制品组成。

在最终部署到生产环境之前，二进制文件穿梭在整个软件交付管道中，通过测试、验证和安全检查点。

虽然每个应用程序都始于源代码，但每个源文件都会通过许多构建和测试步骤生成并依赖于许多制品（二进制文件和依赖项），然后生成可交付的软件。

DevOps 旨在**加快高质量软件发布版本的交付速度。**通过消除软件交付流程中的摩擦来实现这一目标，这在软件生命周期过程交汇处极为常见。此外，一致性和确定性也助力实现快速交付的自动化。



制品生态系统

您的制品生态系统——可部署的二进制文件及其相关文件——处于开发与交付之间。JFrog Platform 控制、简化和自动化二进制文件流程的能力是成功开发 DevOps 最重要的因素。

1 DEVOPS 的唯一可信源

如果所有开发都使用单一的编程语言，遵循一组最佳实践将易如反掌。
但随着各组织将更多的语言和技术添加到工具链中，实践和执行相同的工作流程变得越来越困难。

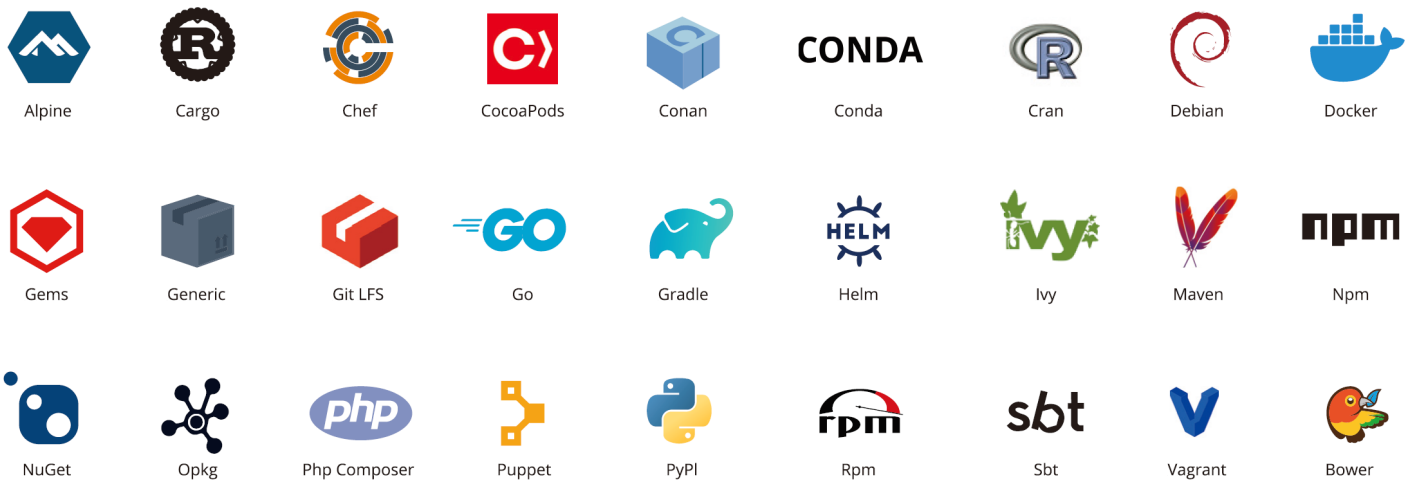
作为 JFrog Platform 背后的力量，Artifactory 是所有制品的通用仓库管理工具：依赖项、二进制文件和配置文件均包含在内。Artifactory 为 30 多种软件包类型（包括通用仓库）提供原生支持，是您在组织开发过程中用于存储、保护和跟踪整个制品生态系统的唯一来源。开发人员可以通过每天使用相同的软件包管理服务，在 Artifactory 仓库中存储和检索软件包或镜像。

摩擦点

多语言开发

大多数企业使用
12 种及以上
不同类型的软件包

超过
50%
的开发人员希望在未来 12 个月内采用一种新的语言



Artifactory 仓库类型

Artifactory 提供的通用二进制管理服务于您组织中的所有开发人员。无论他们是用 Java、JavaScript、Python、Go、C++，还是 C#、Swift、Rust 等开发程序，Artifactory 是所有软件包和构建的中心。

当所有二进制文件都通过这一通用工具进行管理时，整个企业都可以遵循相同的 SDLC 工作流程和最佳实践，从而确保质量并加快发布速度。这就是 Artifactory 成为 JFrog DevOps 平台全自动软件分发流水线核心组件的原因。

2 跟踪软件物料清单 [SBOM]

摩擦点

制品数量

每个机构的制品超过 5 百万

每年制品数量平均增加 130%

构建轻而易举。了解所有构建却并非易事。尤其是每天或每小时都有很多新构建时更是如此。

Artifactory 将新的元数据——我们称之为“构建信息”——与每个构建存储在一起，不仅能链接到您的开源和专有依赖项的软件包元数据，还能链接到构建制品和环境设置。有了详细的构建信息，您可以**追溯每一个构建，从哪里来，准备到哪里服务。**

构建信息是软件物料清单 (SBOM) 的基础，适用于投入生产或交付给客户的每个版本。软件物料清单机器可读，其中详细列出了应用程序中包含的所有项目及其来源。
随着越来越多的政府和受监管行业需要软件物料清单来帮助打击网络攻击，JFrog Platform 是您实现合规性的整体解决方案。
Artifactory 的构建信息有助于您了解构建来源、创建方式或部署位置。

ARTIFACTORY 查询语言

元数据对于 CI/CD 流程至关重要；有助于自动化做出明智的决策。Artifactory 查询语言 (AQL) 使您能够发现与存储在 Artifactory 仓库中的制品和构建相关的所有数据。它的语法用简单方式来制定复杂查询，指定任意数量的搜索标准、筛选、排序选项和输出参数。AQL 作为公开的 RESTful API，使用数据流来提供输出的数据，实现极快的响应时间和极低的内存消耗。

DevOps 工程师可以使用 AQL，在不断增长的仓库中快速识别相关的软件包和构建。他们可以在 CI/CD 服务器中采用 AQL，使原本需要人工干预的流程实现自动化，或者在分析中提高工作流程的性能。

3 通过代理缓存远程仓库

从 npm、Maven、Conan 等远程资源库中提取的开源软件 (OSS) 依赖项极易成为当今应用程序中更重要的代码部分。确保网站的可靠性和快速访问是保障发布速度的关键，但可能面临几项挑战：

网络延迟
物理距离的固有延迟——远程站点可能位于地球的另一端。

负载过重
由于某项服务需求量大导致延迟。

站点停机时间
存储依赖项的远程站点可能会因故障、受到攻击或服务丢失而遭受服务中断。

连接性差
网络中断、抖动、带宽不佳、连接不稳定。

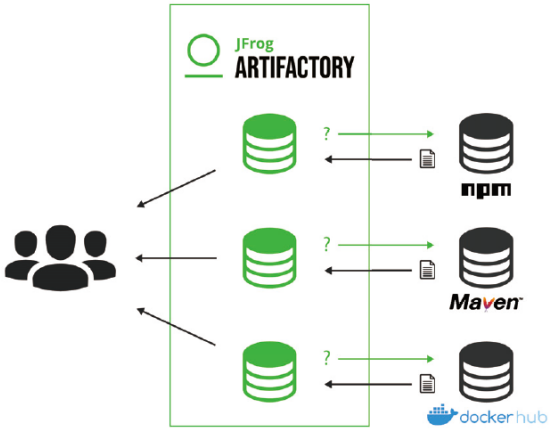
摩擦点

远程仓库

主 npm 注册表中的软件包超过 130 万

92%
的应用程序使用开源代码

Artifactory 中的远程仓库是一个本地代理，在保存本地仓库的同一站点上缓存远程资源依赖项。开发人员无需访问远程资源，而是使用 Artifactory 中依赖项的按需副本进行构建。



这有助于消除因物理距离或不稳定的服务连接所导致的固有网络延迟，并保证构建尽快运行。该代理还可以防止由于连接中断或远程站点本身不可用而造成的中断。
同样重要的是，Artifactory 中的缓存有助于将依赖项保持为不可变的版本——在第一次从远程资源按需拉取之后，缓存中的软件包永远不会改变。防止对远程资源的任何强制覆盖（可能带有恶意），并保证构建的确定性。

虚拟仓库

虚拟仓库可以封装任意数量的本地和远程仓库，并将它们表示为通过单个 URL 访问的统一仓库。借助虚拟仓库，您可以管理开发人员对各个仓库的访问权限，因为您可以自由混合、匹配和修改虚拟仓库中包含的实际仓库。
为了优化制品解析，您还可以定义访问基础仓库的顺序，以便 Artifactory 首先查看本地仓库，再查看远程仓库缓存，再然后，Artifactory 才会直接向远程资源请求制品。对于开发者来说，这很简单。只需发出软件包请求，Artifactory 就会根据您组织的政策以安全且最佳的方式获取它。

4 一次构建，层层晋级

应用持续集成的敏捷方法，每个新的软件版本都必须通过软件开发生命周期中的几道质量关卡。在发布最终版本之前，候选版本经过不同团队的集成、测试和试运行，不断进行晋级。

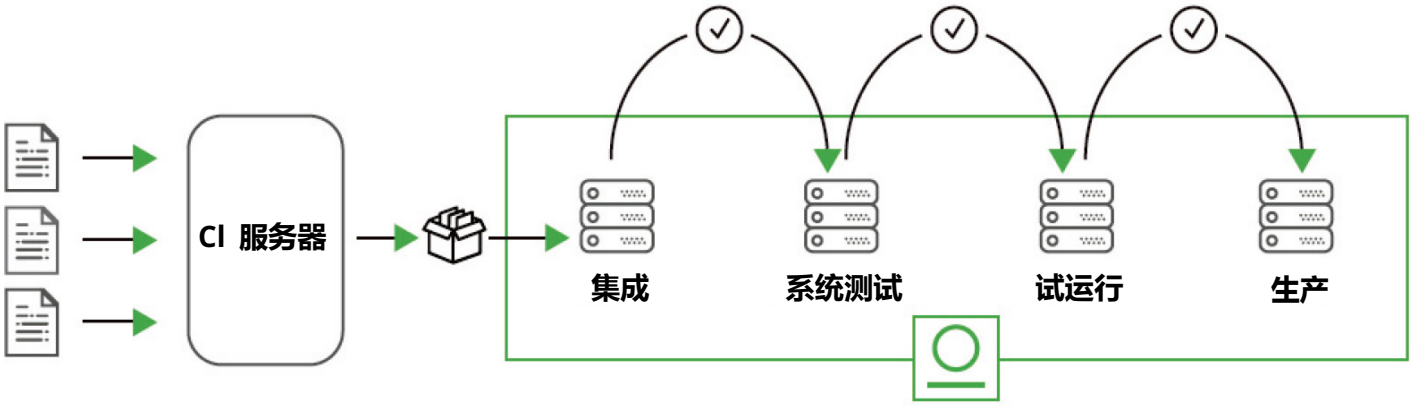
但是通过这些质量关卡的内容决定了是快速发布还是延缓发布。当通过源代码进行版本晋级时，每个接收团队必须在自身的运行环境中执行代码的确定性构建，而这可能会导致不同的二进制文件出现。在每个阶段评估的软件与最后一个阶段缺乏完全相同的确定性。

摩擦点

持续集成

超过
61%
的开发人员每天只需花
2-4 小时进行编码

Artifactory 的二进制仓库管理提供了更一致、更可靠的方法，**在整个软件开发生命周期中只产生一个不可变的二进制文件。**在软件开发生命周期的每个阶段都有一个仓库，在 JFrog Platform 中，只需将带有元数据的构建转移到下一个 repo，就可以进行晋级。



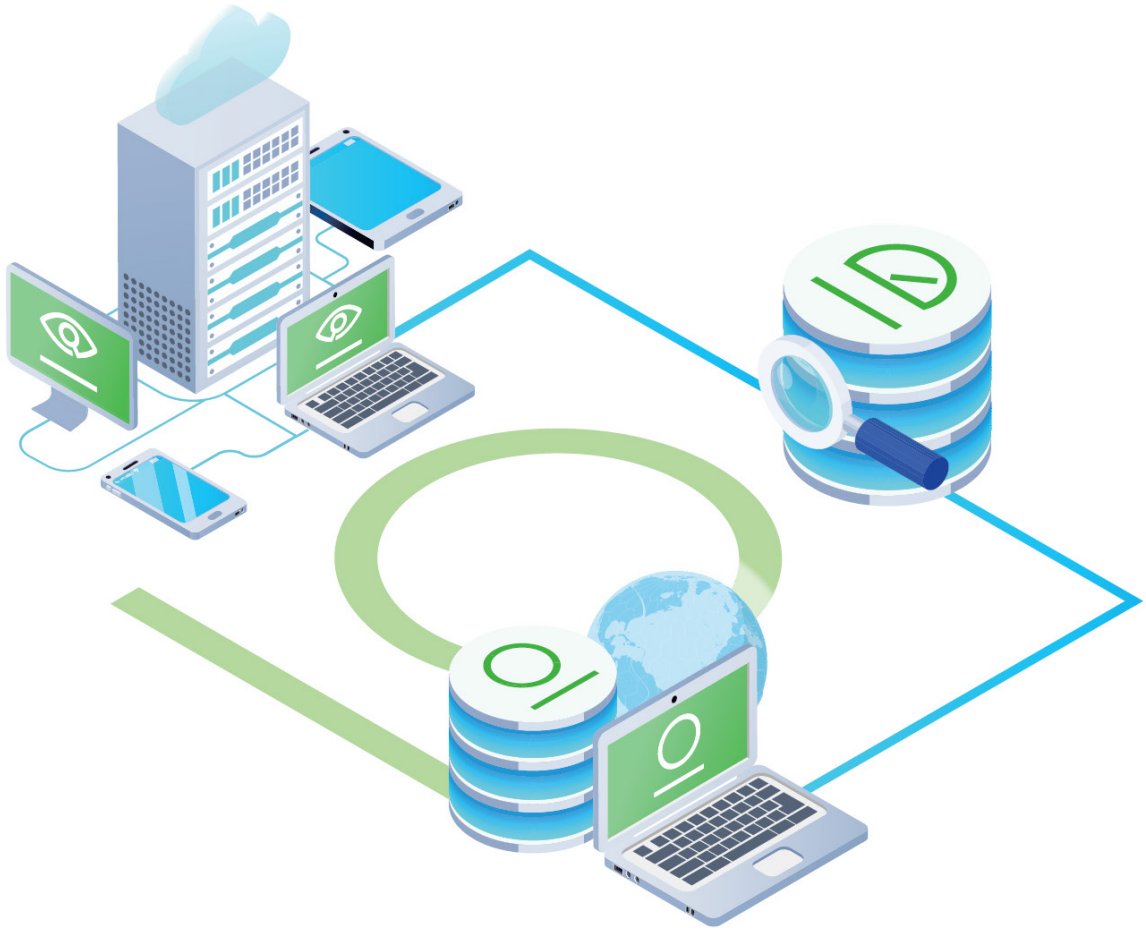
在这种“构建一次并晋级”的方法中，同一构建在每个阶段都要接受评估，以确保整个 DevOps 流水线的绝对一致性。

一旦团队不必执行自己的构建或不必管理构建环境，便可将恢复时间用于执行更详尽的测试并更快提供反馈。

在每个阶段，团队都可以向备选版本的构建信息中添加稳定性、安全性等方面的更多元数据。通过传递从一个阶段学到的、可供下一阶段使用的关键元数据，JFrog Platform 积累了**与投入生产的每个版本生命周期相关的全面记录。**

以“校验和”为基础的存储

JFrog Platform 确保任何二进制文件在文件系统中只存储一次，以此来优化存储。二进制文件存储在本地或远程仓库中时，Artifactory 会计算文件的唯一校验和（支持 MD5 和 SHA1），并将文件重命名为对应的校验和。仓库只保存对文件及其元数据的引用，因此当一个二进制文件被复制或晋级到另一个仓库时，只有引用内容会被更改。物理文件永远不会被复制，它的校验和可以用来验证二进制文件的完整性。



5 加速云原生开发

越来越多的软件开发是云原生开发：编写的应用程序可以有效地利用云技术基础架构，并支持在云端运行的固有最佳特性。

这意味着生产基于容器的微服务，依赖于现有云原生标准，如 OCI 和云原生工具（如 Docker 和 Kubernetes）。

JFrog Platform 中的 Docker 仓库完全支持所有 Docker 镜像中心 API，因此可以与 Docker CLI 一起在本机运行。您可以使用本地仓库，按需在 Artifactory 中维护任意数量的私有 Docker 镜像中心，以便在组织内分发、共享容器镜像。

摩擦点

云现代化

80% 的边缘应用程序部署在 Kubernetes

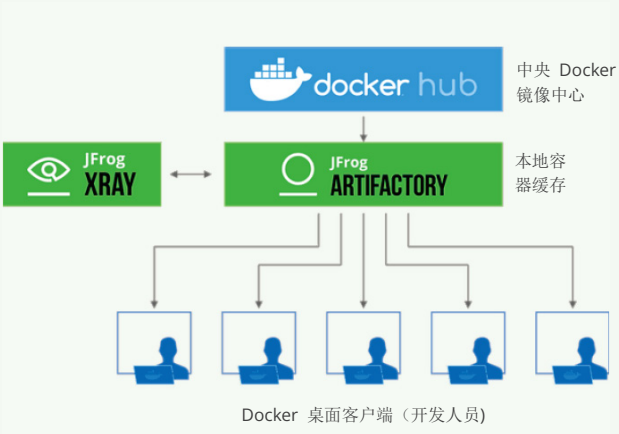
截至 2025 年，全球每年在云应用程序现代化方面的支出超过 240 亿

Artifactory 揭示了 组成 Docker 或 OCI 镜像的每一个层，并将元数据链接到完全可追溯的路径，以追溯到所有部件的起源。
与 JFrog Platform 的精细访问控制相结合，您可以维护比 Docker 可信镜像中心提供的安全性更高的安全私有 Docker 仓库。使用 Artifactory 的本地仓库替代 Docker Hub 上的私有仓库，可避免所有互联网连接问题，对镜像的访问更可靠、更一致。
Artifactory 还支持 Helm Chart 仓库，因此您还可以管理 Kubernetes 编排清单以及 Docker 镜像。通过这种方式，JFrog Platform 可以充当全面的 Kubernetes 镜像中心，是部署到集群中的所有内容的可跟踪主目录。

DOCKER HUB 不受限制

JFrog 与 Docker 的合作使 Artifactory 的 JFrog 云用户不受 Docker Hub 镜像拉取率的限制。通常，匿名免费用户每 6 小时仅可拉取 100 次，认证免费用户每 6 小时 仅可拉取 200 次。但是当镜像拉取请求来自 JFrog 云帐户时，Docker Hub 会免除这些限制。

通过设置 Artifactory 远程仓库来代理 Docker Hub，云用户可以不受限、高性能地访问 Docker Hub 和 Docker 官方镜像，从而简化云原生应用程序的开发。



开发人员还可以利用 JFrog Xray，对从 Docker Hub 拉取的镜像进行持续、全面的漏洞扫描。

6 保护您的软件供应链

软件开发就像烘焙蛋糕。代码是食谱，二进制文件是入口的食材。经营面包店时，顾客从来不会看食谱，但期望食材是新鲜、纯净的。

经营企业时，保护软件供应链（应用程序的食材）免受错误和攻击，对于保护您最珍视的业务来说至关重要。Tyupkin（2014 年）、NotPetya（2017 年）、Operation ShadowHammer（2019 年）和 SolarWinds（2020 年）等供应链攻击提高了人们的认识，促使世界经济论坛将其列为首要的网络安全挑战。作为所有二进制文件的唯一可信源，二进制仓库管理工具是供应链网络攻击中极具吸引力的目标。

摩擦点

网络安全

一家普通公司平均每年有 130 个安全漏洞

美国每年因恶意网络活动造成的损失高达 1090 亿美元

这便是 JFrog Platform 将安全性作为重中之重的原因。

身份验证
JFrog Platform 的所有服务都需要通过安全凭证（如密码或访问令牌）进行身份验证。

校验和验证
每个制品的计算校验和是其在 Artifactory 仓库中存储方式的一部分，用于验证其完整性。

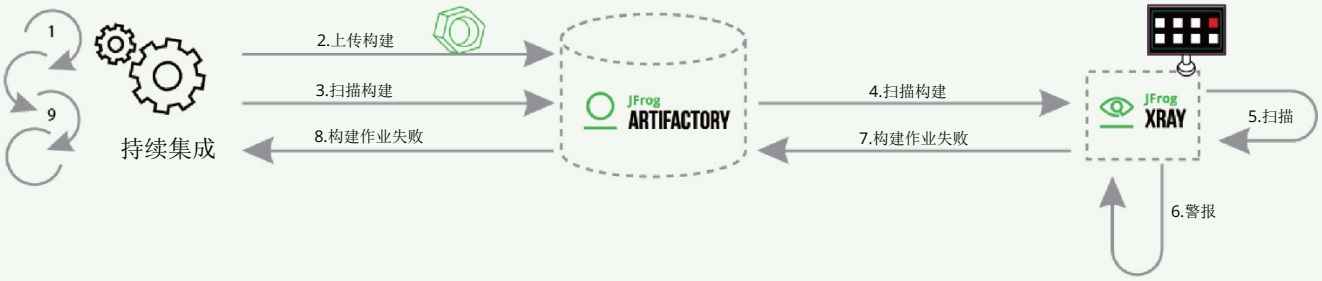
权限管理
通过 JFrog Platform 的精细访问控制，管理员可以确保开发人员和小组只能通过被授权的 CRUD 操作来访问仓库。

单点登录
支持 LDAP、SAML、OAuth 和 SCIM 协议，管理员能够与单点登录服务（如 Active Directory、Crowd 等服务）相集成，在整个组织范围内实现安全的凭据管理。

漏洞扫描

在最近的一项调查中，只有 18% 的组织表示对他们的开源组件非常有信心，这些组件通常占应用程序代码的 60-80%。近两成的人信心不足，或是完全没有信心。

作为配套的安全解决方案，JFrog Xray 对 JFrog Platform 仓库中的二进制文件执行深度递归扫描，以便识别所有具有已知漏洞的开源组件。作为 JFrog Platform 的一部分，Xray 与 Artifactory 紧密集成，维护额外的安全元数据并提供影响分析，因此可快速修复所有使用易受攻击依赖项的二进制文件。



使用 Xray，可防止将高风险的构建部署到生产中，并能召回新发现的具有风险的构建。

此外，Xray 还能监控开源组件的许可证类型，提醒您所有不符合组织策略的情况。

7 连接到您的工具环境

对于开发人员来说，DevOps 是沿同一方向、产品质量达到可发布水平的流水线。

但是对于站点可靠性工程师来说，DevOps 是一个组织结构，将工具和程序编织在一起，打造强大、有弹性的基础架构。

能够很好地集成复杂的 DevOps 工具栈，对于运营的可靠性至关重要，保持生态系统的运行会消耗过多的工程时间。

从构建自动化工具开始：CI/CD 服务器必须与存储它们的系统顺利集成。JFrog Platform 可以通过以下方式连接到所选的自动化 CI/CD：

摩擦点

工具堆栈集成

用于运营的 SRE 时间应少于 50%

CLI

命令行界面 (CLI) 工具使开发人员能够通过命令窗口、shell 脚本或 CI 流水线在仓库中存储、检索二进制文件和元数据。

REST APIs

开发人员可以通过 curl 或定制的 DevOps 工具使用 REST 命令管理二进制仓库。

Webhooks

触发另一个服务中的操作以响应 Artifactory 中的事件，通知用户该事件或启动自动流程。

构建集成

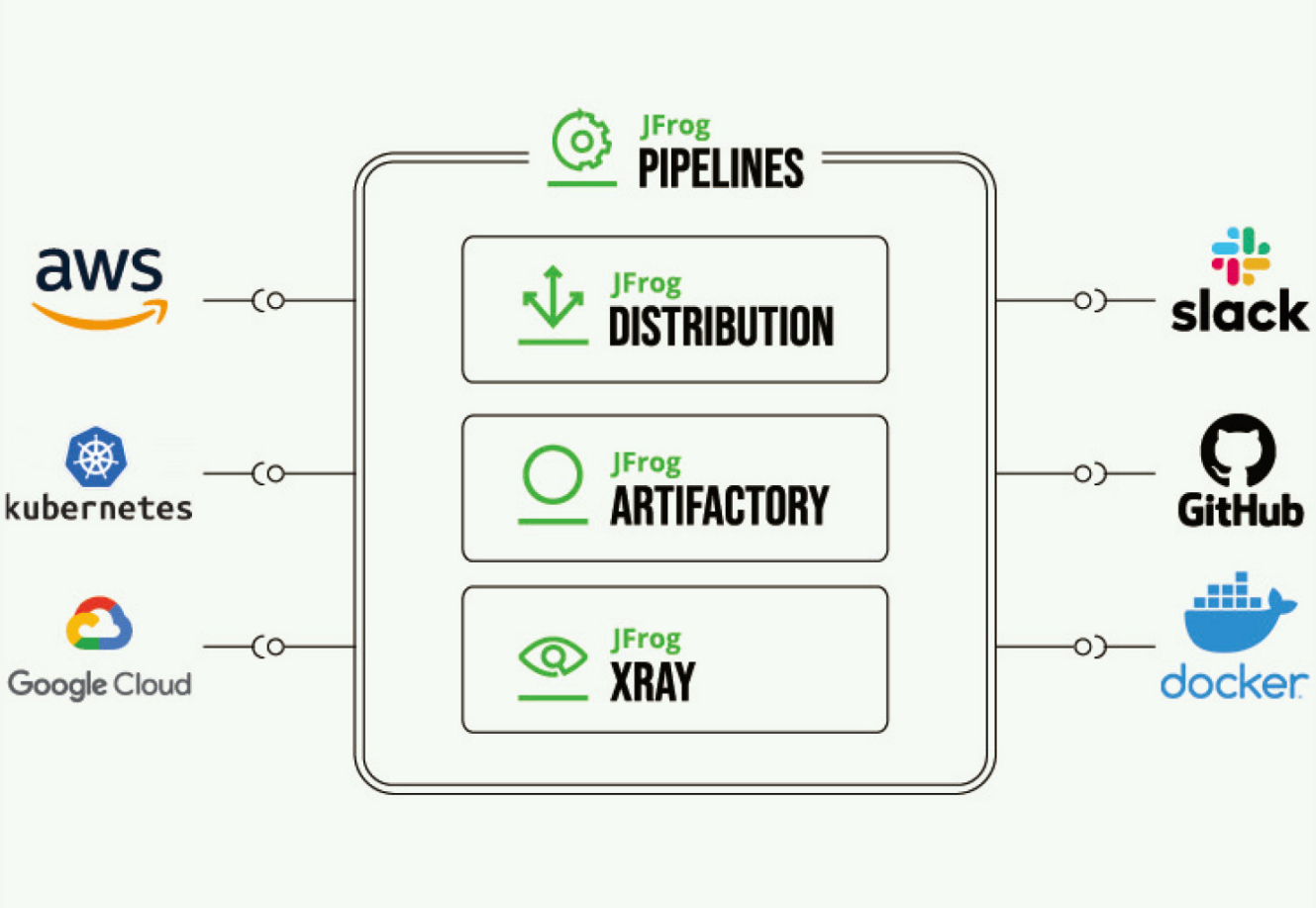
Jenkins、Circle CI、TeamCity、Bitbucket、Pipelines 和 Azure DevOps 等主要 CI 工具的现有插件和扩展可加速与 CI 流水线的集成。

当然，这仅仅是集成需求的开始。用于**自动化测试、协作、ITSM、可观察性和分析**的工具都是 DevOps 基础架构的组成部分。您会希望将这些工具产生的信息推送到仓库中，并将这些信息连接到其他工具。随着 JFrog 合作方与行业顶级供应商的合作，通过唯一可信源，JFrog Platform 可帮助您快速构建强大而紧密的 DevOps 工具栈。

自然集成的 CI/CD

JFrog Pipelines CI/CD 是构建 CI/CD 生态系统的最快方式。作为 JFrog DevOps 平台的一部分，Pipelines 与 Artifactory 自然相结合。使用 Pipelines 的声明性预构建步骤，您可以专注于进入仓库的内容，而无需顾及进入的方式。

JFrog Platform、Xray 和 Distribution 的其余组件，作为 DevOps 领域最为核心的统一工具在安装时就已经预先集成。



Pipelines 还为生态系统中最受欢迎的工具提供现成的集成，包括 GitHub、GitLab、BitBucket、Slack、Jira、AWS、GCP、Docker、Kubernetes 等等。将这些服务中的大多数与 Pipelines 集成，只需输入 URL 端点和用户凭证即可。

8 无限扩展

一个开发小组在同一个房间里开发应用程序的日子已经一去不复返。企业软件开发现在是一项高度协作的工作，由遍布全球多个站点的交叉团队共享软件包。

即使小团队也应该期待成长，开发 DevOps 的最佳实践必须与之无缝衔接。

事实证明，这些以二进制为中心的 DevOps 最佳实践（唯一可信源、元数据、构建晋级、安全性等）可以顺利扩展。每天，JFrog Platform 的用户都能快速发布高质量软件，无论是由 5 个人在一个房间里开发，还是由全球 500 个人共同开发。

作为完成任务的关键工具，即便您的部门在发展壮大，也需要确保持续、快速的访问。

摩擦点

增长和规模

每个企业平均有 5 台带有 Artifactory 的服务器

每个企业帐户的二进制文件超过 1100 万

截至 2023 年
全球软件开发商超过 2700 万

高可用性

在自有本地安全数据中心的服务器上自行托管，或在自有的云帐户中作为集群中的 BYOL 安装自行管理。

集群中多活节点的高可用性配置有助于分散负载，适应突发的大负载情况，确保没有单点故障。最大限度地延长了正常运行时间，甚至在系统更新期间，也可达到 99.999% 的可用性水平。

云 (SaaS)

JFrog 云管理服务可托管在您选择的主要云服务提供商 (AWS、GCP 或 Azure) 平台上，订购后可“随时随地”利用云端的可用性和弹性。采用多云策略，维护托管在不同云服务提供商上的多个 SaaS 帐户。有助于避免供应商锁定，并支持将工作负载分配给最具成本效益的提供商。

混合云

混合云策略将您自有防火墙背后的自托管服务器与云中的托管服务相结合。您可以将云的可扩展性用于动态工作负载，同时将敏感工作负载保存在安全的本地数据中心内。

JFrog Platform 承诺在每个环境中都具有相同的功能，这意味着您可以随心所欲地在云和本地之间分配工作负载。

跨地域多站点同步

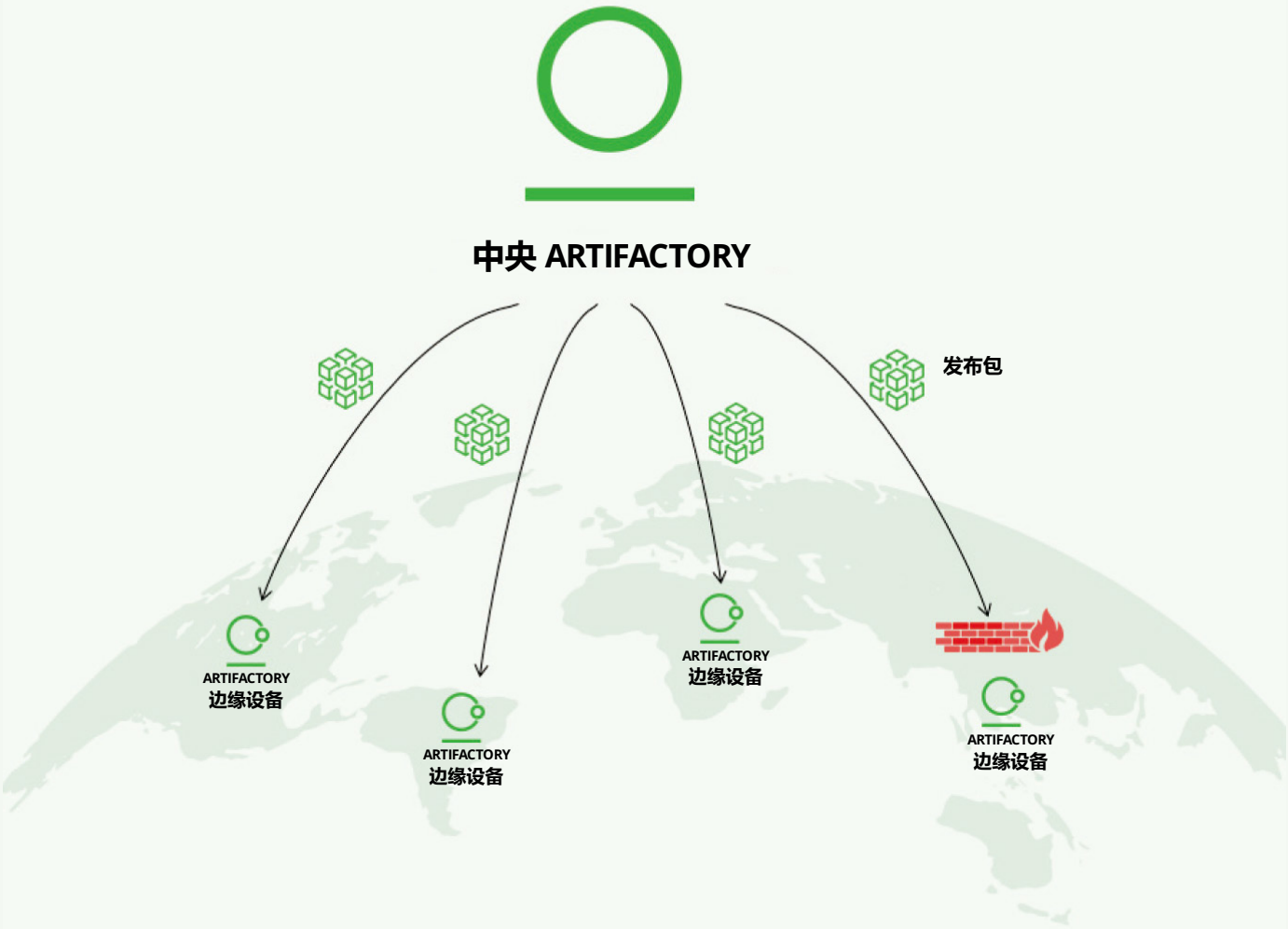
通过几个推送/拉取复制拓扑选项或双向联邦仓库，JFrog Platform 支持跨地域多站点的同步。

JFrog 独特的多站点同步功能确保在全球任何网络拓扑中均可提供本地化访问。这使得地理上分散的团队能以最小的延迟处理相同的制品（二进制文件及其元数据），以便每个站点的每个构建都可以快速完成，不会失败。

分发到最远的边缘设备

一旦拥有完全有效的发布版本，需要将它放在何处？企业必须将软件和更新分发到越来越多的全球端点，交付给集群、设备和桌面，从而使用户受益。

作为 JFrog DevOps 平台的一部分，JFrog Distribution 使 DevOps 团队能够高效地打包来自 Artifactory 的发布包（二进制文件、制品和元数据），并自动将可信软件交付到全球数十乃至数千个远程站点。使用 JFrog 边缘节点——经济高效的只读 Artifactory 实例——您可以通过私有数据网络传输经过签名的、不可篡改的发布版本，这些版本在每个目的地都经过验证，同时保持精细的访问控制。



ARTIFACTORY 让您的二进制文件发挥作用

显然，成功开发 DevOps 的关键在于在整个软件开发生命周期里有效管理二进制文件，尽可能地减少或消除流水线中的摩擦。二进制制品仓库管理工具是数字化转型的关键代理。

我们在这里给出的 8 大理由是以二进制文件为中心的 DevOps 开发方法的基本实践。它们也是 JFrog Platform 构建的核心价值，为您的成功提供更好保障。

Artifactory 的设计和功能使开发人员能够采用 DevOps 的最佳实践，减少生命周期关键阶段的摩擦，加快软件交付。

摩擦点

增长和规模

超过 8300 家组织使用 JFrog

超过 70% 的《财富》500 强公司使用 JFrog

超过 420 亿的制品存储于 JFrog Artifactory 仓库

数以千计的 JFrog 客户时刻都在证明，这种以二进制文件为中心的敏捷数字化转型方法极为有效。

JFrog Platform

Artifactory 是支持 JFrog DevOps 平台的核心组件，该平台是面向一站式 DevOps 的全面端到端平台解决方案。

JFrog Platform 有多种本地部署**订购级别**，或作为云托管服务提供。每个级别都包括一组满足您所在组织需求的 JFrog 组件。无论您是内部应用程序的小型开发团队，还是提供安全边缘软件的全球性企业，JFrog 均可提供适合您的解决方案。

统计数据和来源

> 到 2023 年，全球数字化转型的总支出将达到 6.8 万亿美元
IDC：数字化转型支出

> 超过 70% 的数字化转型工作未能实现目标
McKinsey：转型的“方式”

> 大多数企业使用 12 种或更多不同的软件包类型（JFrog 内部数据）

> 超过 50% 的开发人员希望在未来 12 个月内采用一种新语言
JetBrains：2020 年开发者生态系统的状态

> 每个组织拥有超过 500 万个制品（JFrog 内部数据）

> 制品数量平均每年增长超过 130%（JFrog 内部数据）

> 2020 年，主 npm 注册表中有超过 130 万个软件包（npm 博客）

> 超过 92% 的应用程序使用开源代码（Tidelift，2018 年开源调查）

超过 61% 的开发人员每天只需花 2-4 小时进行编码
ActiveState：2019 年开发者调查

> 超过 80% 的边缘应用部署在 Kubernetes 上

> 到 2025 年，全球每年在云应用程序现代化方面的支出超过 240 亿美元
研究和市场，应用现代化服务市场……2025 年全球预测

> 普通公司每年平均发生 130 起安全违规事件
Ponemon Institute：2017 年网络犯罪成本

> 在美国，每年因恶意网络活动造成的损失高达 1090 亿美元
美国经济顾问委员会：恶意网络活动给美国经济带来的成本，2018 年

> 用于运营的 SRE 时间应少于 50%
谷歌：网站可靠性工程

> 平均每个企业有 5 个搭载 Artifactory 的服务器（JFrog 内部数据）

> 每个企业帐户有超过 1100 万个二进制文件（JFrog 内部数据）

> 到 2023 年，全球有 2700 多万名软件开发人员

Evans Data Corp.：2019 年全球开发人员数量统计研究