



SOFTWARE ARTIFACT STATE OF THE UNION:

JANUARY 2023

The Leading Software Components in Use Today to Inform your 2023 Projects

Copyright © 2023 JFrog Ltd.

Published: February 2023

INTRODUCTION TO SOFTWARE ARTIFACT STATE OF THE UNION

Tools like Stack Overflow's Developer Survey and the Tiobe Index are helpful assets for IT and software leaders to understand the latest development trends and programming design preferences to consider when launching new initiatives. These reports can also be helpful for developers or DevOps engineers to reference when determining how they'd like to upskill themselves or their teams. However, such reports rely only on indirect data such as surveys, GitHub repos, or Google trends, which doesn't necessarily provide a full picture of what languages are actually running in production and/or enterprise ready.

JFrog is in a unique position to detail the actual technologies being used to create software consumed by end users today due to our unmatched depth and breadth of package support and large customer base. With over 7K customers worldwide, spanning single users to the largest enterprises, including the majority of the Fortune 100, JFrog's Software Artifact State of the Union provides a reliable snapshot of package popularity and adoption trends.

WHY ARE SOFTWARE PACKAGES IMPORTANT?

Looking at programming language popularity is one indicator of developer preferences and trends – but packages and binaries are the true components being used by enterprises to deliver software from design to production. Looking at software packages is also a fairly reliable proxy for programming languages given that most package types only serve one or two languages at most.

METHODOLOGY

The data used for these rankings and analysis was collected between October 2021 to October 2022 and based on usage of [JFrog Artifactory](#) by:



Package popularity is based on the total number of repositories maintained as well as the actions taken for a given package type.

It's possible that a handful of enterprises could skew these rankings by creating an unlimited number of repositories for any given package, however, because we also look at artifact actions we can safely conclude which package type is actively being used as part of the development process.



FINDINGS

Top Software Technologies

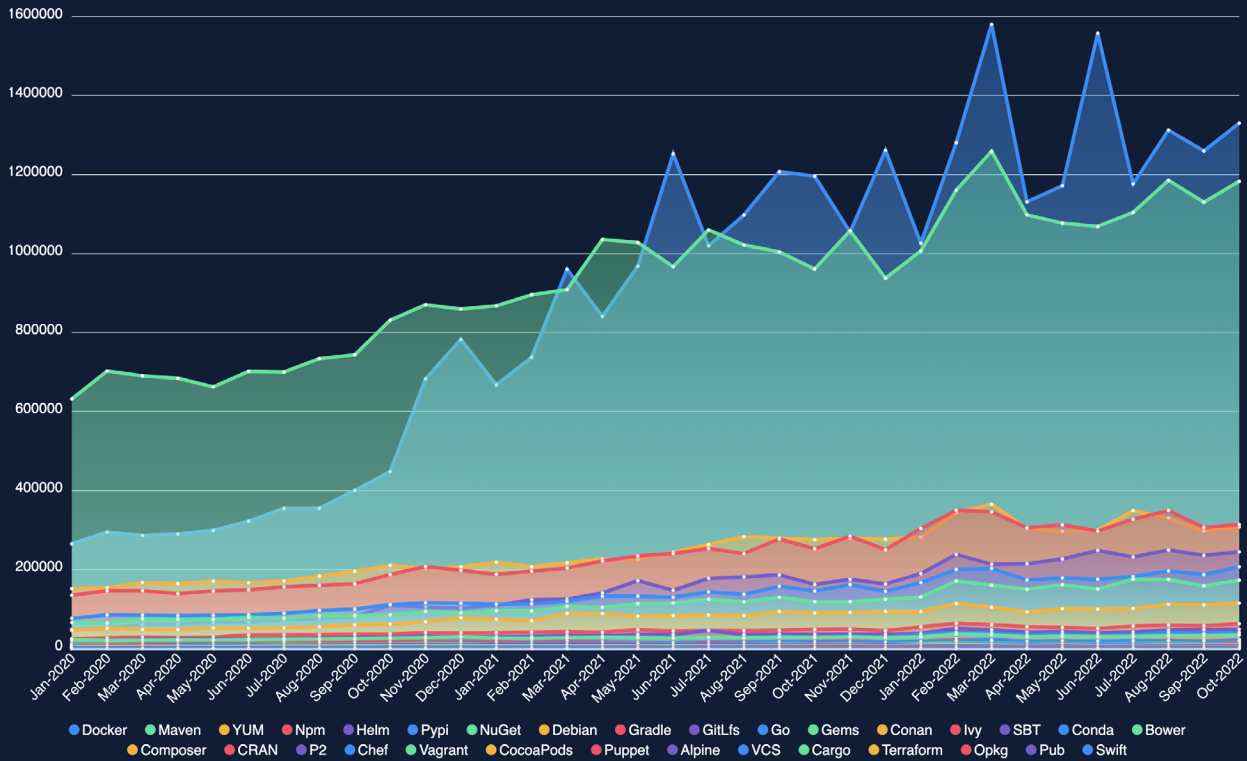
Rank	Package Type	Repo Count	% YoY Growth
1	Docker	1,330,329	10.11%
2	Maven	1,183,167	18.79%
3	Npm	313,992	19.61%
4	YUM	307,549	10.36%
5	Helm	244,582	33.67%
6	Pypi	206,830	29.51%
7	NuGet	172,989	31.54%
8	Debian	114,737	21.56%
9	Gradle	62,842	22.80%
10	GitLfs	48,734	26.34%
11	Go	40,481	29.10%
12	Gems	34,292	17.93%
13	Conan	23,423	29.64%
14	Ivy	21,537	19.90%
15	Conda	20,247	23.78%
16	SBT	17,262	16.82%
17	Composer	12,909	38.74%
18	CRAN	12,151	36.15%
19	Bower	12,124	8.36%
20	P2	10,771	5.48%
21	Chef	10,547	14.27%
22	CocoaPods	9,011	28.39%
23	Vagrant	8,424	11.53%
24	Puppet	6,360	21.97%
25	Alpine	5,535	49.81%
26	VCS	4,961	18.48%
27	Cargo	4,205	67.13%
28	Opkg	2993	32.31%

This table reflects popularity based on total number of repositories for a given package/technology type based on a comparison of October 2022 to the prior year.

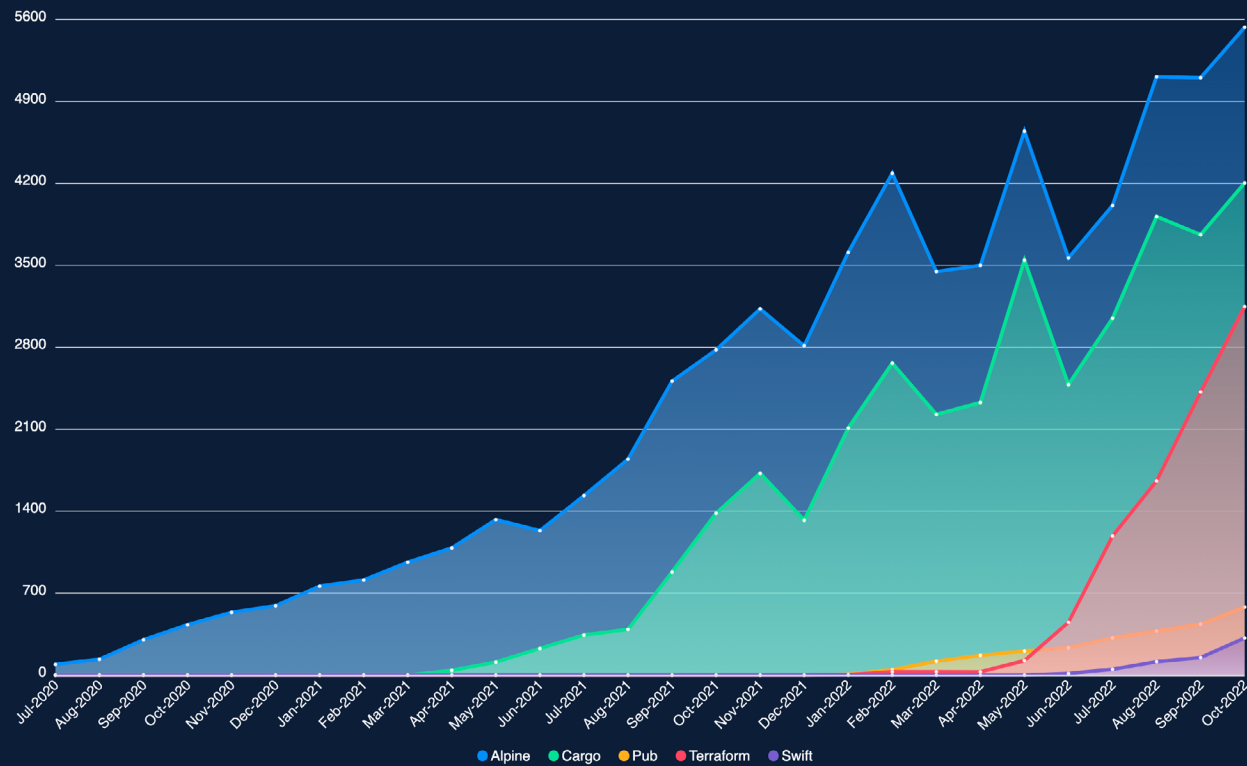


Software Technology Trends

By Popularity



Recently Added to Artifactory



Containers are King



Containerized applications have transformed the way we bundle and deliver software to end users. The rapid rise in use of Docker plus OCI containers and Helm Charts (5x growth from January 2020 to October 2022 respectively) illustrates how common it is for organizations to take a cloud-native approach to **DevOps**.

While Kubernetes has been around since 2013 it's only recently started gaining steam as a solution for deploying containers in large organizations — and it's still maturing at this point – so stay tuned for further developments.

The Old Guard Stands Strong



While there is a lot of interest around younger programming languages such as Rust (Cargo), you can't deny the role of traditional languages like Java, JavaScript, Python, and C/C++ in modern software development. For example, our data shows over 90 percent of orgs maintain a Maven repo, clearly indicating organizations aren't abandoning the use of traditional languages.

Preparing for IoT and the Edge



The number of connected devices is expected to grow to 41.6 billion in 2025 generating more than 74.9 Zettabytes of data according to IDC (**Internet of Things and data placement**). Organizations need a strategy for delivering software to devices at the edge. C/C++ is the primary language used when designing software to run on IoT devices as most are microcontroller-based. Conan is a commonly used C/C++ deployment manager that makes it easier to deliver updates to edge and IoT devices because it is packaged-based as opposed to using a standard dependency library management. This helps expedite the speed and consistency of IoT device software development.

From January 2020 to December 2022 we saw a 5x increase in Conan utilization possibly indicating more companies are designing for the edge.

Keeping Memory Safe



Poor memory management has been the cause of many vulnerabilities. Even the US Government's National Security Agency is urging organizations to move towards memory-safe languages when building applications. A security forward design is just one of the many reasons why developers are interested in leveraging younger languages such as Rust (Cargo), Swift, and Go. For example, Rust is inherently built with memory management as the standard, and it also attributes to have more safety with "zero bugs" which means it is designed to make the developer aware of any potential issues when coding. However, **even Rust has its challenges** and needs to be monitored on a continual basis.

The number of Rust (Cargo) repositories has increased 30% from January '22 to October '22. Modern languages, such as Rust, Swift, and Go are designed to be more accountable by providing better package style development with built-in safety mechanisms. It's possible the desire to work with memory-safe language was also driven by the widespread impact and lessons learned by Log4Shell, Solar Winds, and other detrimental software supply chain attacks.

Terraform: The Standard for IaC



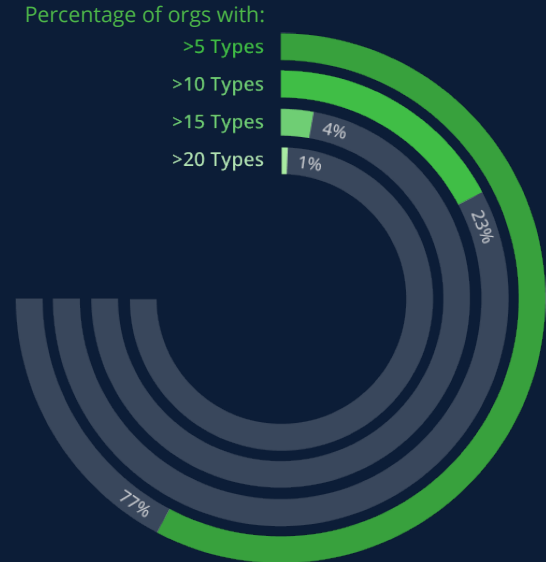
Since introducing support in May, we've seen a rapid adoption of Terraform repositories—one of the fastest growth trajectories of any new package type we've introduced. Organizations see the benefit of moving management of these files out of Git and other storage options and into a fully featured binary based management solution that can keep their Terraform files secure and readily available alongside the other components required to deploy and run their software.

BONUS INSIGHTS ON PACKAGE UTILIZATION TRENDS IN DEVELOPMENT ORGANIZATIONS

In addition to package popularity among developers, JFrog wanted to examine what package utilization typically looked like in development organizations –i.e., how many package types are leveraged, the variety and size of artifacts, plus the average number of developers within any organization. Here’s some additional insights based on that data.

Development is Polyglot and Multi-tech

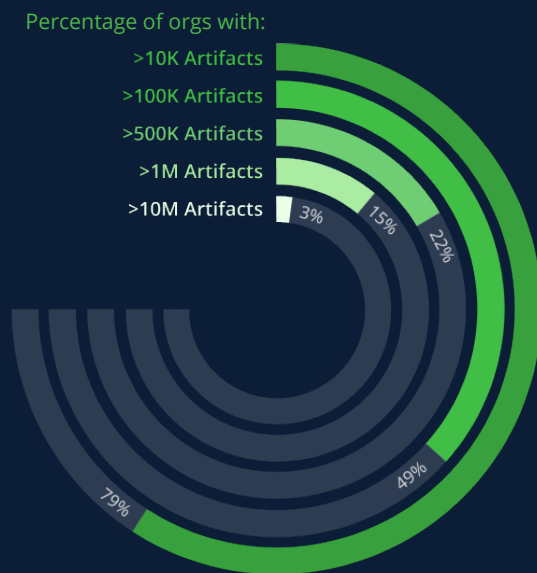
Today’s extended software supply chain involves multiple technologies and languages to deliver applications across on-premises, cloud, and hybrid environments. Drawing from a wide sample of companies - of varying sizes and cross-industry – there’s typically seven or more unique package types in use on average. That median increases within larger organizations, tapping out at 29 different package types used within a single company.



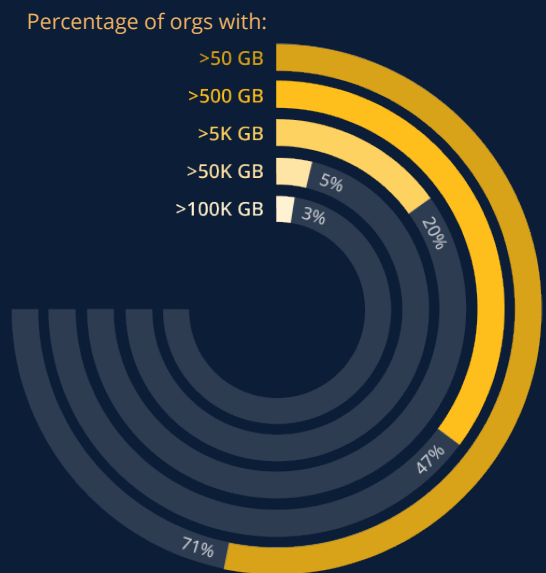
Number of package types used in software delivery

Higher Volume of Larger Artifacts

With the exponential rise of orgs using containerization technologies like Docker and Kubernetes, it’s no surprise to see artifact size increase as well. Additionally, a higher percentage of larger organizations are also maintaining a wider array of artifacts.



Total artifacts managed



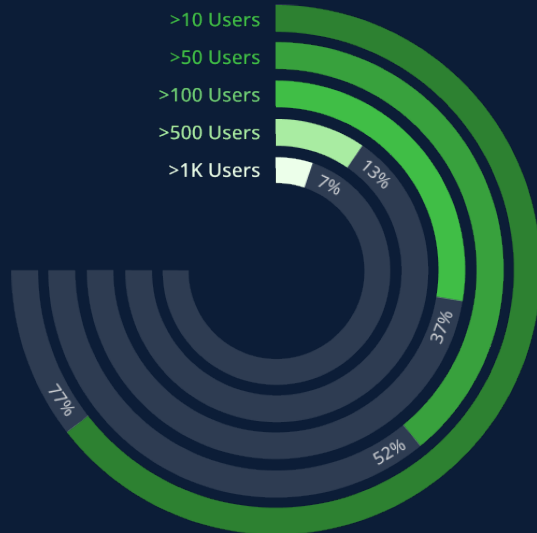
Total artifacts managed by GB



More Users, More Repos

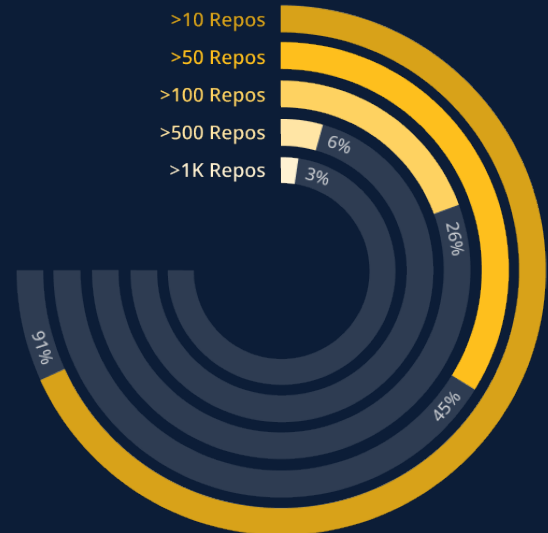
As organizations grow they tend to leverage more repos to keep their artifacts organized and secure. A proper repository strategy with governed role-based access controls helps keep development teams aligned and ensures on-demand access to widely used, common software components.

Percentage of orgs with:



Size of development orgs

Percentage of orgs with:



Number of repositories

PREDICTIONS FOR SOFTWARE PACKAGE TRENDS IN 2023

There's no denying the pandemic accelerated all industries' migration to the cloud, which correspondingly increased use of cloud-native technologies like Kubernetes (K8s) and containerization methods like Docker. Additionally, now that employees are going back to the office and more citizens get back out 'on the road', we'll continue to see increased use of the cloud, collaboration tools, and a growing number of devices being used both remotely and while we're on the go.

This trend will continue fueling the push for use of cloud native technologies, which puts increased pressure on developers to deliver new, mobile-first features and functions rapidly, while ensuring they are always secure and up-to-date. Gartner research indicates more than 85 percent of organizations will embrace a cloud-first principle by 2025 and will not be able to fully execute on their digital strategies without the use of cloud-native architectures and technologies.¹ Don't be left in the dust by your competition.

¹ <https://www.gartner.com/en/newsroom/press-releases/2021-11-10-gartner-says-cloud-will-be-the-centerpiece-of-new-digital-experiences>



From a talent standpoint, this might mean you need a few extra developers on your UX, cloud native, or mobile-first design teams. While there are still a few industries where migration to the cloud is slower due to business and/or data location, data privacy/sovereignty, regulatory requirements, etc., the overall shift to the cloud and adoption of K8s will continue to accelerate as the variety, volume, and size of software packages continues to grow. We'll also likely see increasing hoards of companies take advantage of the dynamic nature of cloud infrastructure with frameworks such as Terraform.

Looking at our own data as well as industry and economic trends, we expect to see growth in Docker and container utilization remain strong given the rise in variety and size of artifacts used to support technologies such as cryptocurrency, metaverse, cloud gaming, and blockchain, among other things.

While the jury is still out on the "metaverse", we see many organizations experimenting in this area with increasing artifact size, the proliferation of C++, Python and Rust (Cargo), along with containerization technologies. At the same time, the increasing use of C++, Python, and Java could also signal more widespread development and use of Blockchain technology which would coincide with the industry's vehement focus on security in general and, more specifically, securing the software supply chain. The popularity of traditional languages like Rust, Java, and JavaScript, also signals a drive by many organizations towards next-generation internet services using decentralized web infrastructures to create new application workflows.

In short, given the many varied use cases enabled by a variety of software languages, companies should be looking for a comprehensive and universal platform that allows them to manage their entire software supply chain from end-to-end, while fueling collaboration, scalability, consistency, and security.

