# 2020 State of Open Source Report

*Market Trends, Analysis, & Looking Ahead*

www.openlogic.com

2020 STATE OF OPEN SOURCE REPORT

# Executive Summary

The 2020 State of Open Source Report from OpenLogic provides data on the most popular open source technologies used today, and sheds light on the barriers and benefits teams experience when adopting open source technologies.

# Foreword

To the reader,

I've been predicting the future for a long time. Even before I founded OpenLogic back in 1998, I thought companies would eventually use open source in critical applications if they could mitigate the associated technical and business risks.

I also anticipated an explosion in the number of open source software (OSS) projects, even though a well-known venture capitalist told me there would eventually be only 10 major OSS packages in the world. Oops.

I wasn't thinking this big at the time, but fast forward 22 years to a world where GitHub has well over 100 million repositories and then consider we're not done growing yet — not by a long shot.

Since OpenLogic created the industry's first enterprise Open Source Census in 2008, we've seen some common results from our community surveys. Support is the number one reason why teams aren't adopting OSS faster. Developers use way more OSS than management thinks they do, especially if they think it's 'none.

Security grows in importance every year — this year it became the most important factor when purchasing software — which is good! OSS quality is good, but many people struggle with documentation, migrations, keeping up with patches, and the "tyranny of choice" between open source options.

For several years, we've also seen JavaScript top the most popular languages list and Python continue to gain momentum, strongly driven by its domination in the AI space. However, all the languages we asked about have significant usage. More than ever, developers are polyglots and choose the best tool for the job. Based on the responses to our language mix question, they care about flexibility and ease of deployment more than anything else when it comes to assembling the right mix of technologies. This shows that user experience applies to developers as much as it does to end users. After all, anything that makes our lives easier when writing code is most welcome!

One surprise I found in the results this year is that only half of the respondents use one of the popular tools for automation and orchestration, such as Ansible, Puppet, Chef, or Terraform. If you fit in the same category, please use automation to accelerate DevOps. You will get better quality and reduce your stress while shipping faster.

I was also pleasantly surprised to see Istio, the open source service mesh, hit the radar for the first time this year. If you have a complex microservices architecture not based on a service mesh, check it out. Among many great benefits, you will avoid tons of duplicated effort and get a more resilient production environment as a result.

Read on to learn more about what your peers are using, what they are avoiding, and where the industry is heading. If there is one easy prediction I can make about the future, it is this: we will see even more open source usage next year, particularly in innovative areas like AI and service mesh.

Enjoy the report,

Rod Cope

Rod Cope

CTO, Perforce Software

# Table of Contents

# About the Survey

The 2020 State of Open Source Report from OpenLogic is based on an anonymous survey conducted between April and July of 2020. The survey received a total of 1085 responses.

The survey focused primarily on open source industry adoption, organizations' software considerations, and the most widely used technologies within:

- Infrastructure and Operation Systems
- Applications and Runtime
- Databases and Analytics
- SDLC and CI/CD
- Automation and Orchestration

## Reporting & Analysis Methodology

To make the report more scannable, we created graphs that disregard outliers. For this report, we considered any question choice with fewer than 1% of respondent choice to be an outlier.

## Respondents

For collecting this survey data, we reached out to IT operations and development professionals with experience in several open source technologies to gain understanding in their software requirements and support needs. This report gives insight into the types of technology organizations are using and their main concerns when it comes to utilizing open source software.

# Survey Results

## Open Source Adoption Rates & Trends

### Is your organization using open source software?

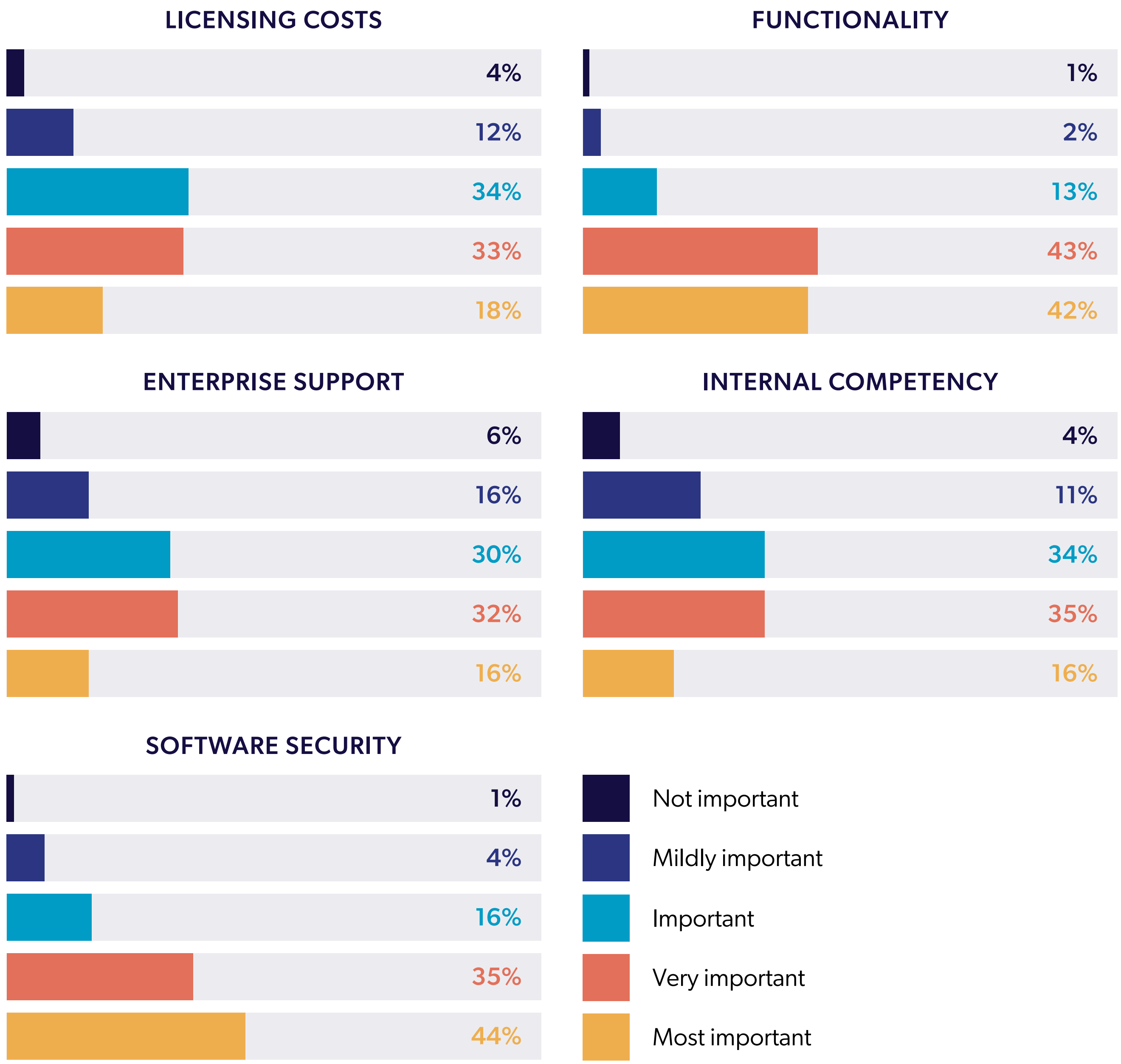
90% of respondents in this survey stated that they are utilizing open source software within their technology stack. Given the skyrocketing rate of open source adoption within commercial organizations within the past few years, this is no surprise.

### Is your organization considering any new open source technologies?

79% of respondents replied that they are considering new technologies within open source. Considering 90% of respondents are already using open source, this reinforces the success that businesses are having with open source software. Organizations are seeing the initial benefits and they are seeking more opportunities to use it.

### Software Purchasing Considerations

In this question, we asked respondents to rank (from not important-most important) their organization's considerations when purchasing software. Considerations included: Licensing Costs, Enterprise Support, Software Security, Functionality, and Internal Competency.



Over 44% of respondents listed software security as the most important consideration in purchasing software, with functionality trailing slightly at 42%.

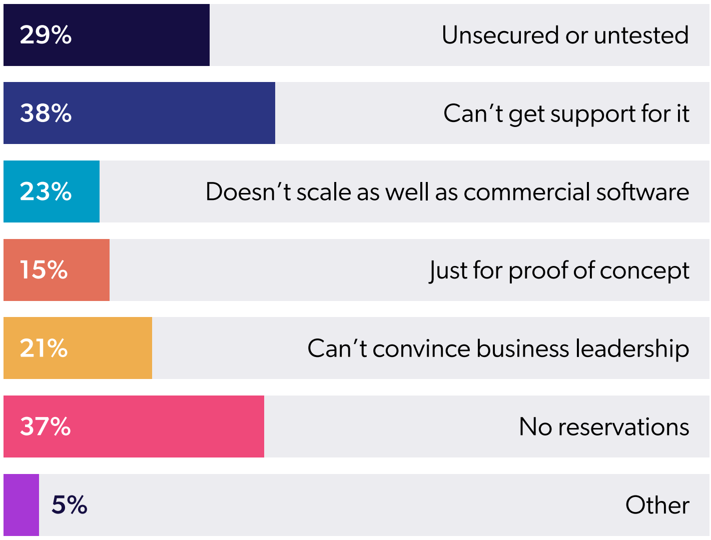
Is this consideration for security an artifact of the long-held mythology that open source is less secure? Or does it represent the importance of software security among development teams? Regardless, keeping data secure is a top reported priority.

Other concerns, like licensing cost, internal competency, and enterprise support were rated less important than security and functionality, on average.

# Reservations for Adopting Open Source Software

Many of our respondents, as noted in previous sections, are already utilizing open source software or are currently considering new software. But that does not mean adoption is coming without certain reservations.

**What are you reservations in migrating to open source software?**



Our survey found that 38% of respondents reported support as a pain point for open source software adoption. An additional 29% reported security and testing concerns as their primary obstacle, while 23% reported concerns with scalability as compared to commercial software options as their primary.

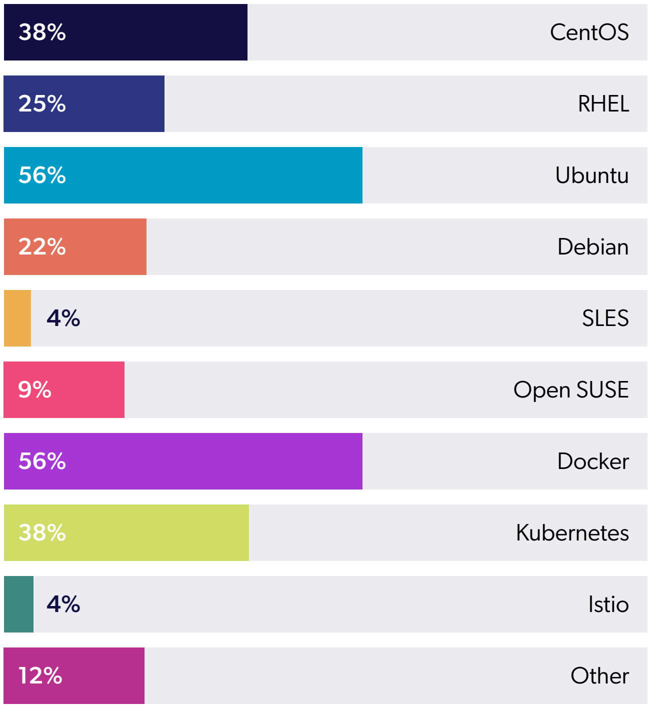
For those who have been in the industry for the last few decades, support has long been the biggest hurdle for adoption. Luckily for developers, OpenLogic provides full enterprise support for over 400 open source technologies, including 24/7 break/fix support and consultative guidance.

Seeing that over a third of respondents report no reservations about adopting open source software shows the impressive trajectory of open source software quality, reliability, and adoption.

The concerns over security and testing are interesting to see, especially considering that the code within typical open source software is always validated and never goes untested. Anecdotally, security and testing accountability for code within open source software is often higher than code within commercial counterparts.

# Top Technologies in Infrastructure

In this question, we asked respondents to list the infrastructure technologies they use to support their software. So what do those results look like in the "year of the container?"



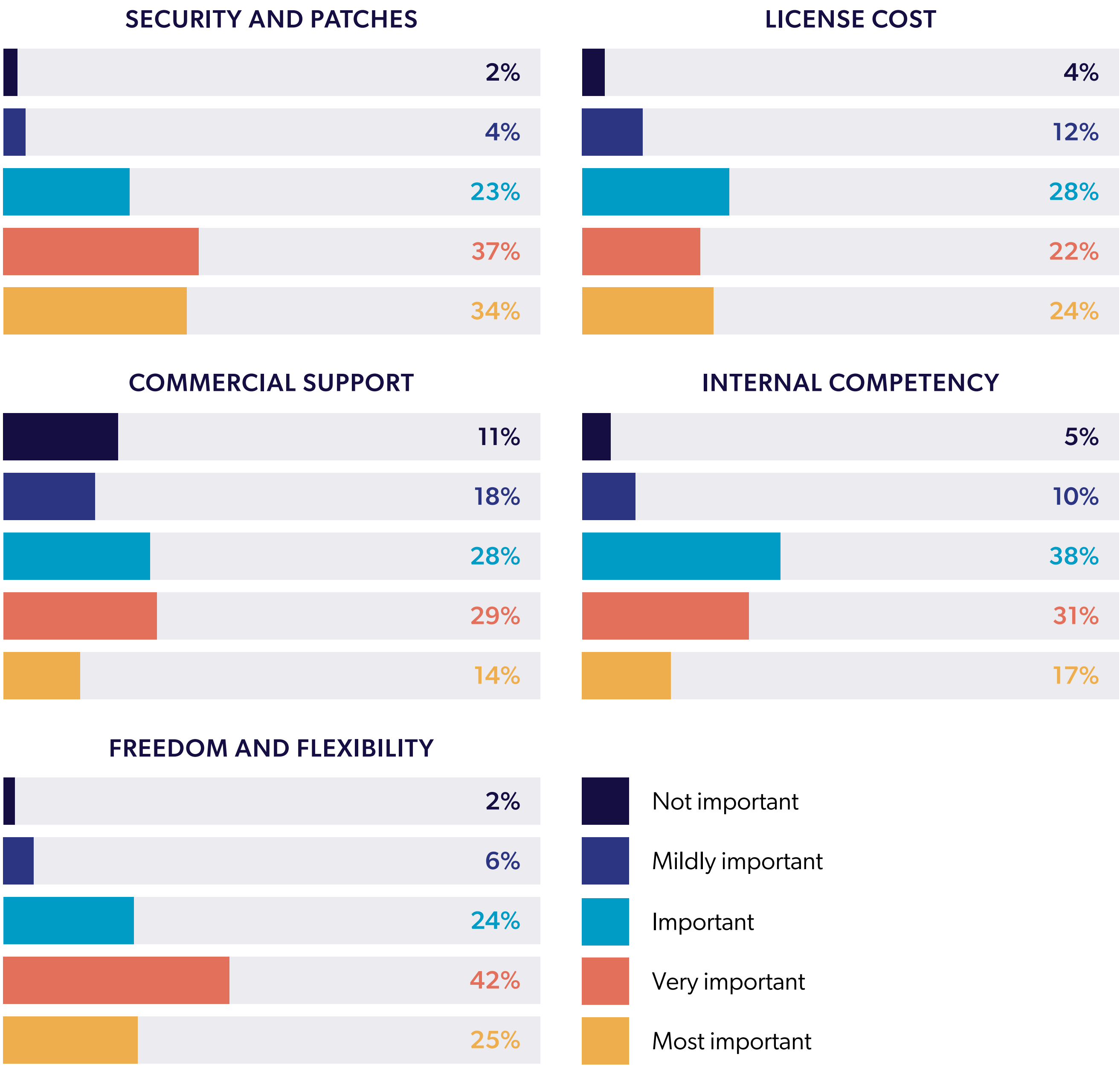
Our survey found that 56% of respondents reported using Docker, with 56% using Ubuntu. Trailing behind the top two technologies, we saw Kubernetes and CentOS at 38% a piece. Moving down the list, RHEL usage showed at just over a quarter of respondents, with Debian, OpenSUSE, and SLES coming in at 22%, 9%, and 4%, respectively. Finally, the open source service mesh technology, Istio, came in at 4%.

There are a few interesting points here, namely Ubuntu outpacing CentOS, and CentOS outpacing RHEL. With many users migrating from RHEL to CentOS to gain infrastructural freedom, that trend is likely to continue.

It's also important to note the emergence of Istio as an open source service mesh option. While Istio adoption isn't on pace with Kubernetes, there is plenty of room for it to grow market share in coming years.

# Infrastructure Considerations

For this question, we asked respondents to tell us what they considered most important when choosing which technology mix to use for their infrastructure. In order of importance (not important – most important) respondents ranked the following reasons: Security and Patches, Commercial Support, Freedom and Flexibility, License Costs, and Internal Competency.



Our survey found that security and patches were the most important point of consideration when choosing infrastructure technologies, with 34% of respondents reporting it as their most important consideration.

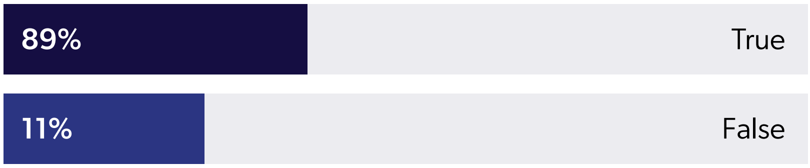
Behind security and patches, freedom and flexibility was the next most popular choice. The weighted average for responses found that freedom and flexibility trailed security and patches only slightly, showing that many respondents found it as the biggest payoff in choosing new infrastructure technologies.

License cost, internal competency, and commercial support trailed these choices, showing security concerns as a continuing theme among teams considering open source technologies.

## Infrastructure – High Availability

Next, we asked respondents to weigh in on their team's ability to keep infrastructure technology highly available in production.

**I feel confident in my team's ability to keep our infrastructure technology highly available in production.**



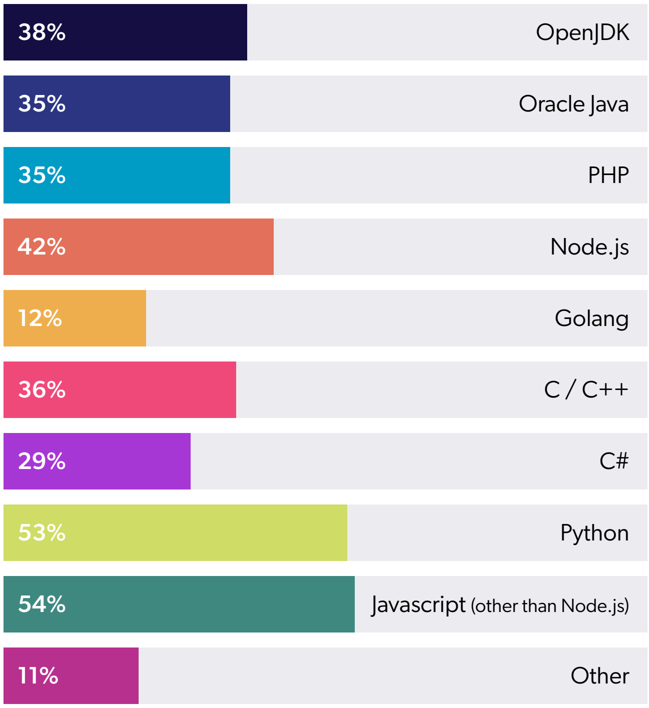
Our survey found nearly 90% of respondents reporting confidence in maintaining high availability in production.

This is a massive change from just a couple years ago, where high availability was a big concern. So what changed?  In the last few years there has been a significant increase in cloud technology adoption, as well as advancements in failover tech, and general software maturity.  With organizations able to offload a lot of their redundancy and failover to cloud services, and software just getting better in general, availability is becoming a solved problem.

# Top Technologies in Applications/Runtime

In this question, we asked respondents to select the technologies they use to build applications.

**Which technologies do you use to build your applications today?**



Our survey found 54% of respondents using Javascript, 53% using Python, 42% using Node.js, and over 38% using OpenJDK.

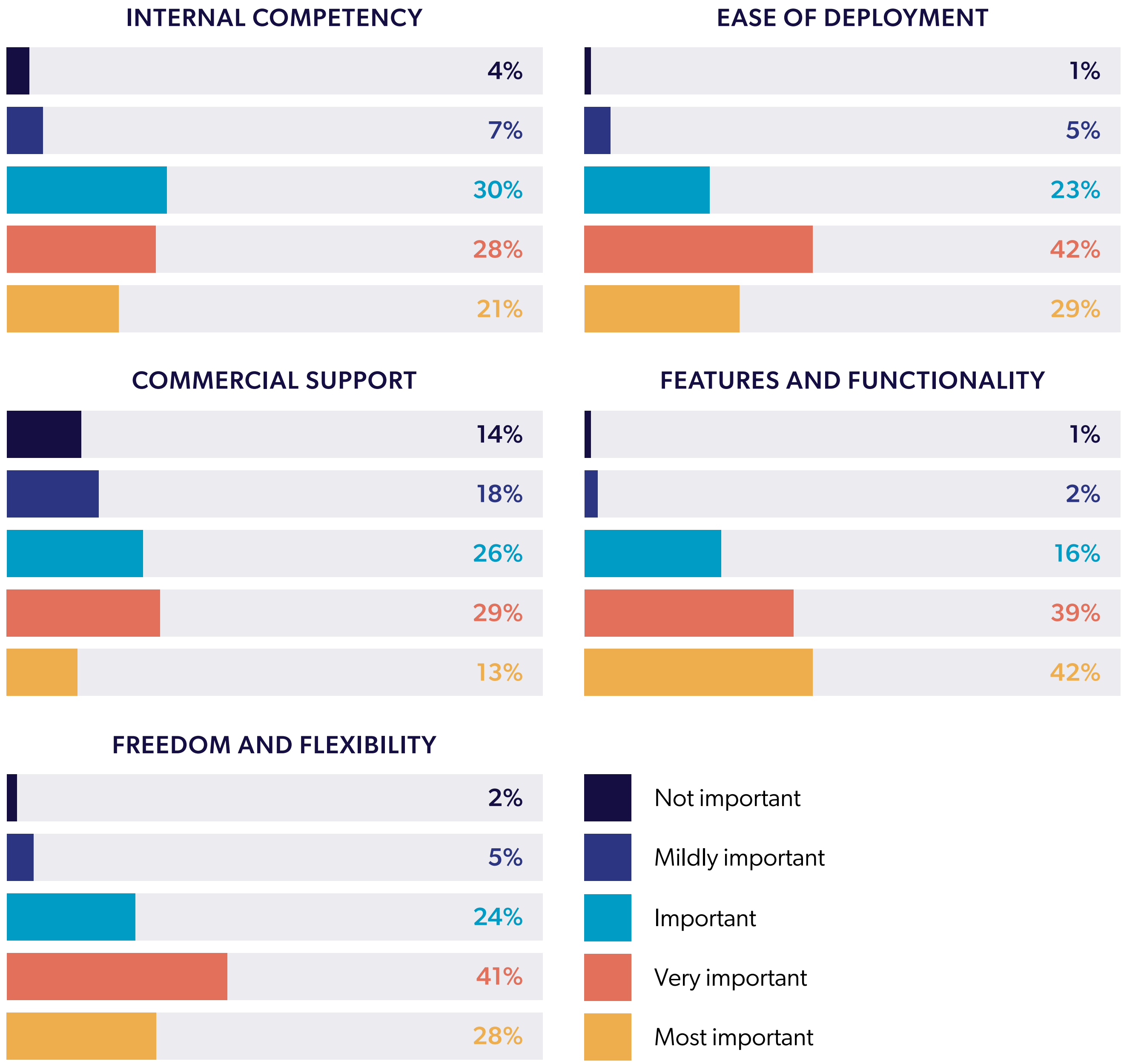
Trailing those, C / C++ users reported in at 35%, and Oracle Java and PHP users at 36% respectively. Lastly, over 28% of respondents reported using C#, with another 12% using Golang.

While it was interesting to see OpenJDK outpacing Oracle Java (as predicted by Gartner), the big surprise is with Python – which fell just short of JavaScript for the most popular language.  New open advancements in AI and scientific / analytics software is likely driving a lot of this adoption, with Python being incredibly popular in those fields in particular.

Nearly every language on this list is in heavy rotation – showing that technologies are more able to support the increasingly polyglottal nature of development.

# Applications/Runtime Considerations

For this question, we asked respondents to tell us in order of importance (not important – most important) the reasons they chose their development technology mix. Reasons provided were: Internal Competency, Commercial Support, Freedom and Flexibility, Ease of Deployment, Features and Functionality.



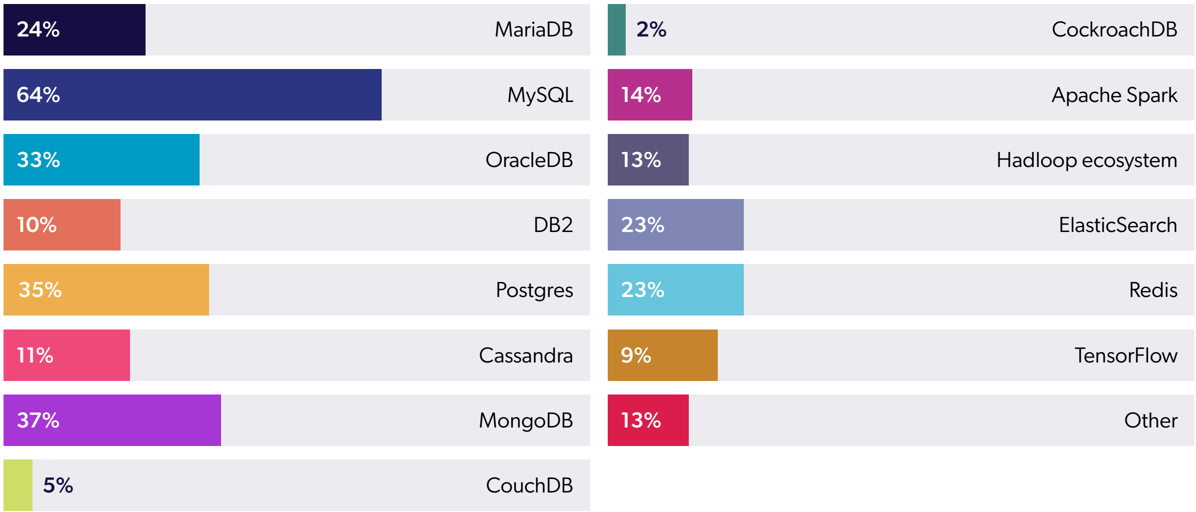
Our survey found that the most important consideration for respondents when choosing development technologies was in features and functionality. Next, ease of deployment and freedom and flexibility were nearly split, with 29% and 28%, respectively. Internal competency and commercial support were the least important considerations, with 21% of respondents reporting internal competency, and 13% reporting commercial support as their most important consideration.

These findings are in line with other responses within the survey, and show a trend that teams put a high value on the functionality, flexibility, and ease of use for development technologies.

# Top Technologies in Databases/Analytics

Our next question asked respondents to weigh in with the database and analytics technologies they use today.

**Which databases and/or analytics technologies are you using today?**

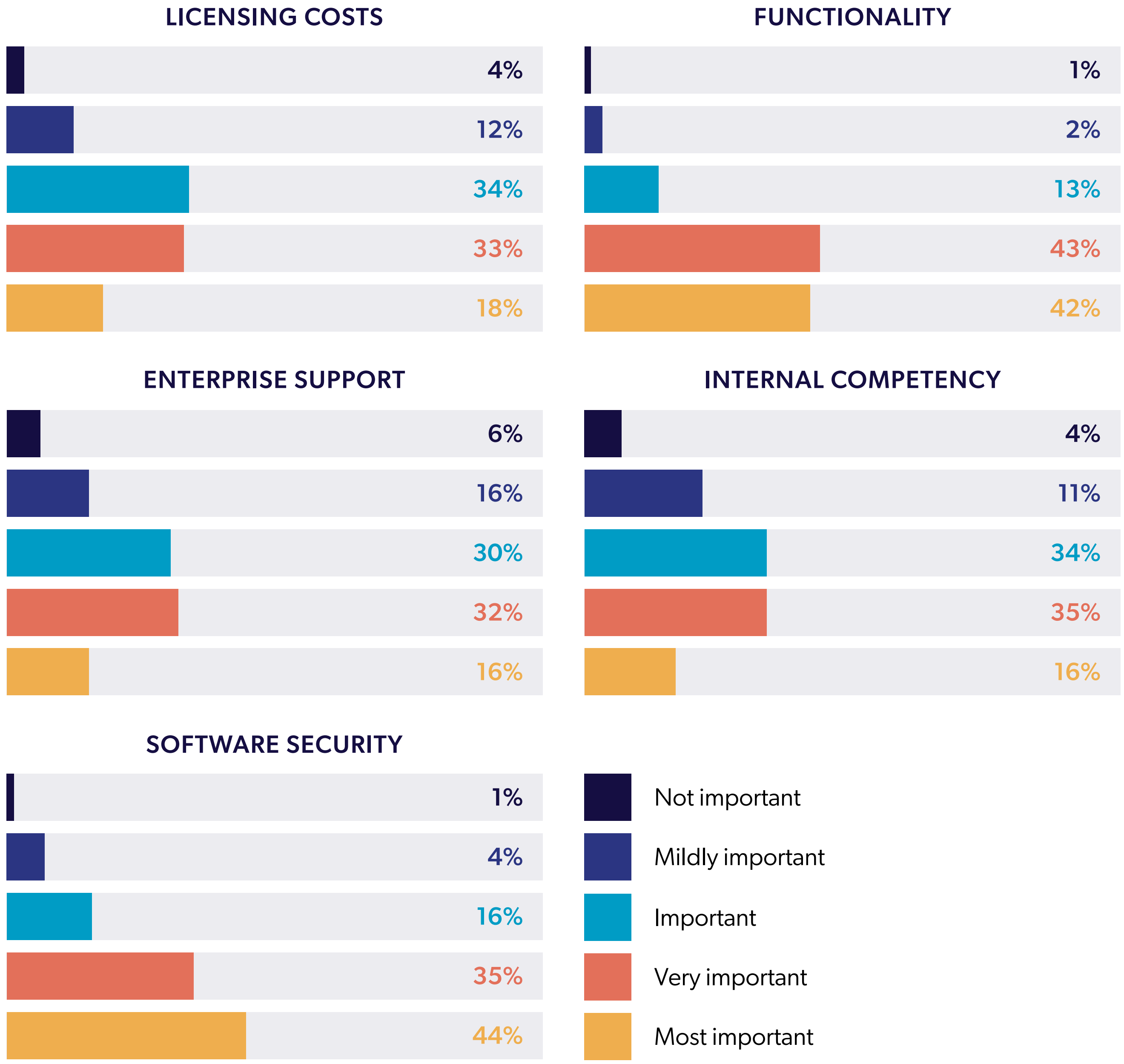


MySQL was by far the most popular choice, with nearly two thirds of respondents reporting using a MySQL database in their current application. Behind MySQL, MongoDB, Oracle DB and MariaDB were the next most popular, at 37%, 33%, and 24% adoption, respectively.

It was interesting to see Postgres outpacing Oracle within this audience, as other surveys have shown Oracle DB well ahead of Postgres. The high level of MySQL adoption shows that despite being an older database technology, it has a lot of staying power.

# Database/Analytics Considerations

In this question, we asked respondents to tell us in order of importance (not important – most important) the reasons they chose their database/analytics technology mix. Reasons provided were: HA / DR Capabilities, Big Data Needs, Scaling for Clients, Ease of Use/Deployment, Features and Functionality.



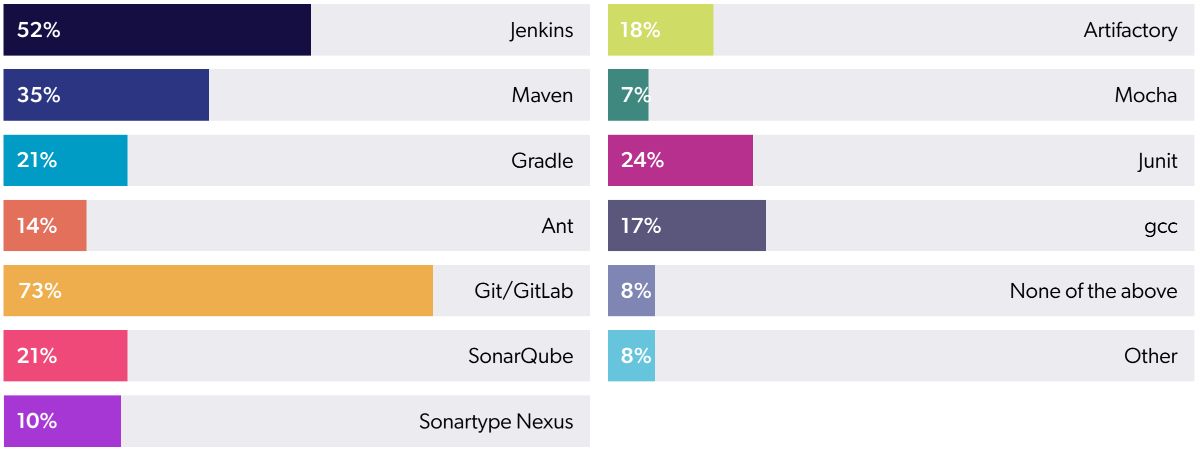
Our survey found that features and functionality were the most important considerations for choosing database and analytics technologies. Ease of use and deployment considerations were also important, followed by scaling, HA / DR capabilities and big data needs.

It is interesting that ease of use was rated so highly for databases and analytics tools. Many analysts don’t necessarily have a strong software background, so having easy to use software often leads to better adoption.

## SDLC + Build Technologies

This question asked respondents to select the SDLC and build technologies they currently use.

**Which SDLC and build technologies are you using today?**

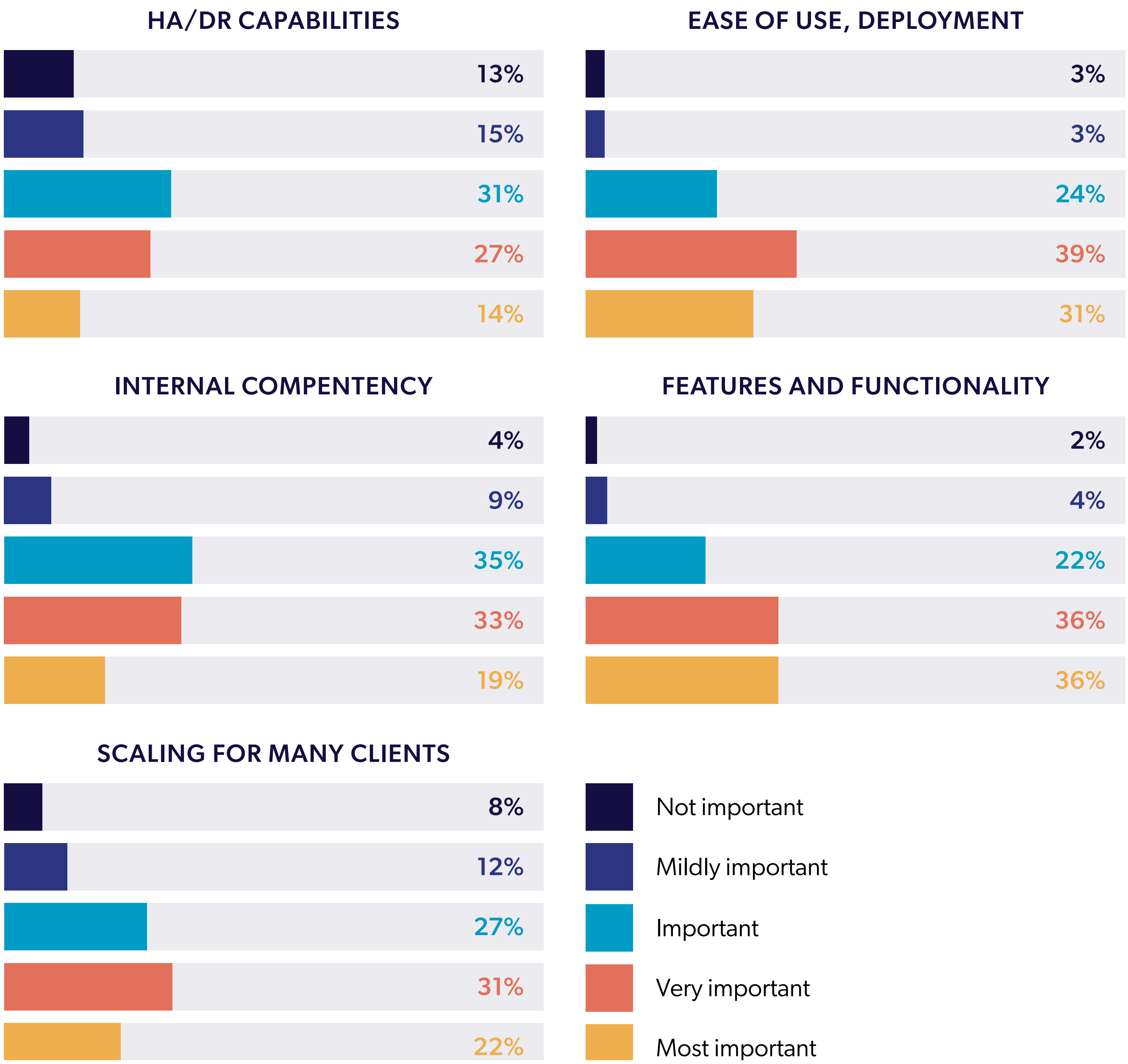


Our survey found the most popular technologies are Git / GitLab at 73%, Jenkins at 52%, and Maven at 35%.

It’s no surprise to see Git and Jenkins at the top of the charts here, though it is surprising to see Maven still edging out Gradle as the most popular Java build tool.

## SDLC + Build Technology Considerations

For this question, we asked respondents to tell us in order of importance (not important – most important) the reasons they chose their SDLC and build technology mix. Reasons provided were: HA / DR Capabilities, Internal Competency, Scaling for Clients, Ease of Use/Deployment, Features and Functionality.



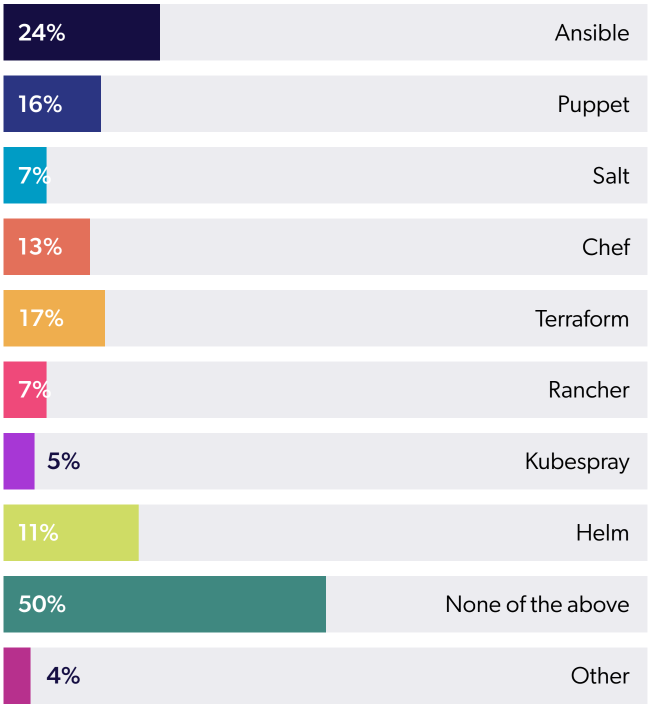
Our survey found that features and functionality and ease of use were the most important factors in choosing SDLC and build technologies. 36% of respondents reported features and functionality as the most important factor, while 31% reported ease of use and deployment considerations as the most important factor.

The wider a feature set of an SDLC stack, the more able it is to deploy to a wide set of environments. That means more cloud compatibility, easier integration with newer deployment infrastructure like Kubernetes, and in general more value for the organization who is adopting the stack.

Internal competency, scaling, and HA / DR capabilities were viewed, on average, as less important factors.

# Automation + Orchestration Technologies

In this question, we asked respondents to select the automation and orchestration technologies they use today.



Our survey found Ansible to be the most popular technology, with nearly a quarter of respondents using the open source software.

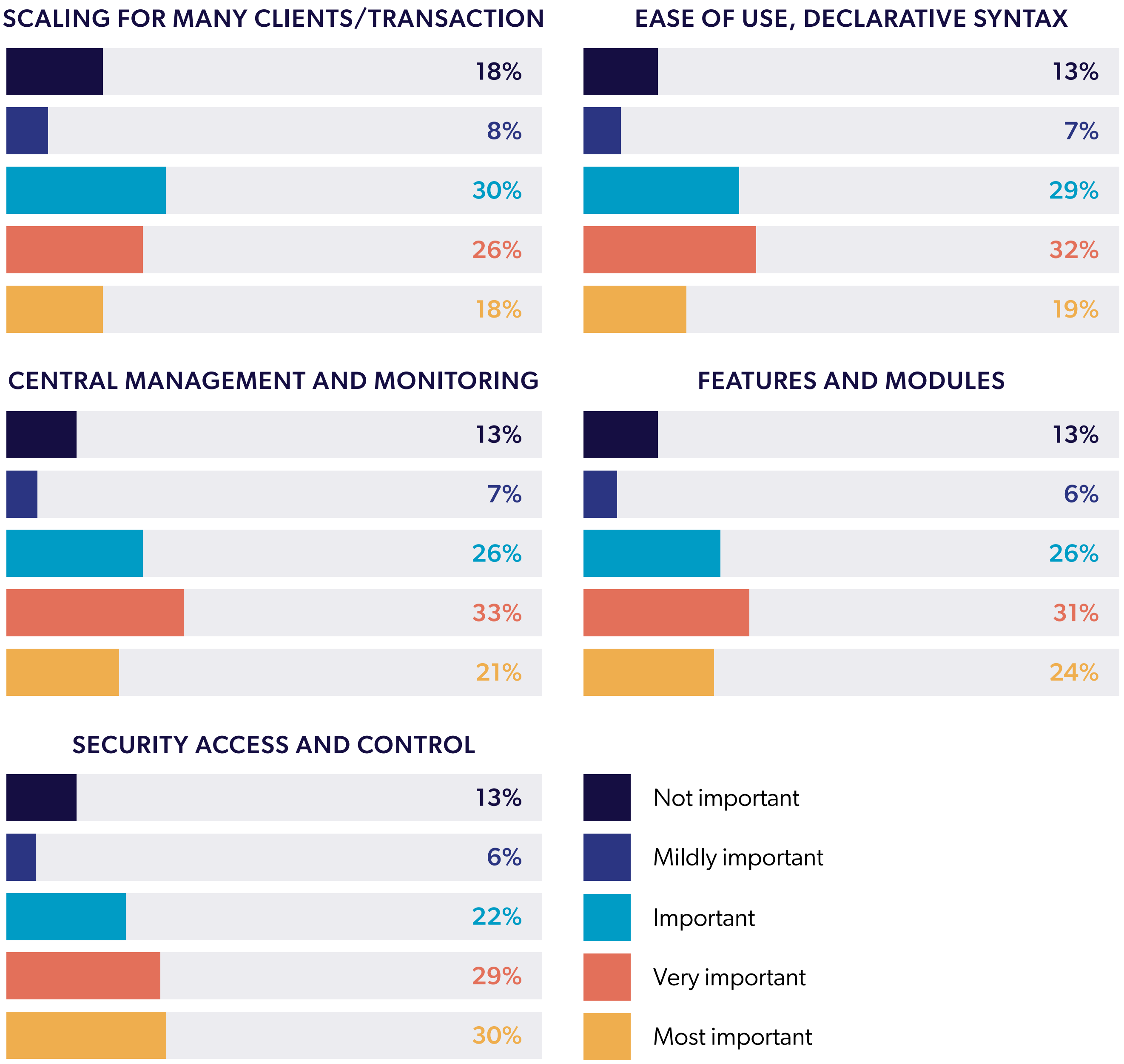
Terraform, Puppet, Chef, and Helm were the only other technologies above 10%, at 17%, 16%, 13%, and 11%, respectively.

50 of respondents reported not using any of the listed, industry-standard tools. What does that mean? It could be that many teams are either doing this work manually, rolling their own tools, or have a big opportunity to improve automation.

With Ansible continuing to grow as a proven, stable, open source orchestration solution, we expect to see more teams adopting it, and realizing the benefits of IT automation and orchestration in coming years.

# Automation + Orchestration Considerations

Our next question asked respondents to select their considerations for automation and orchestration technology adoption.

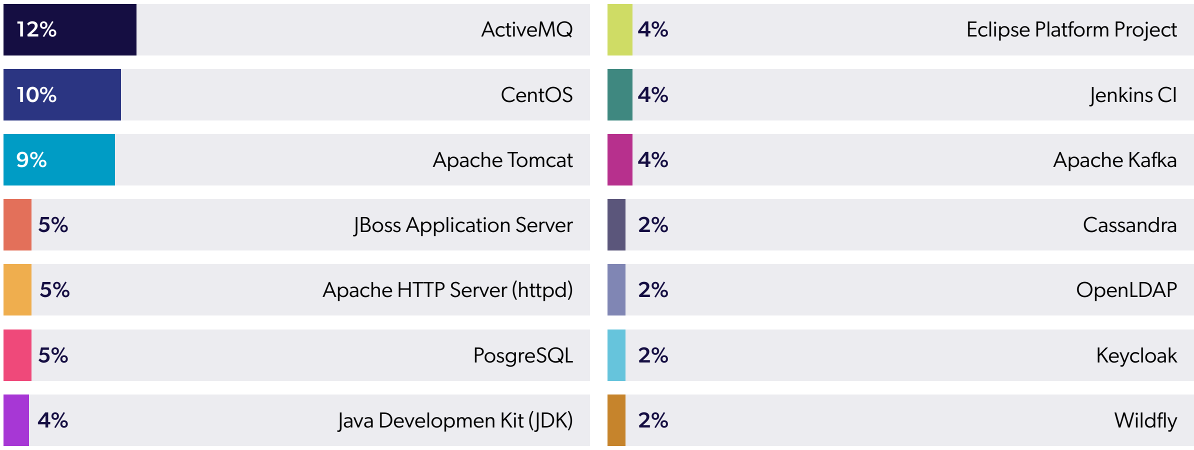


Our survey found that the most important factor for choosing automation and orchestration technologies was in security and access control, followed by features, management, and ease of use. Scaling was reported as the least important factor.

The results point to an acknowledgement of the devastating consequences of a compromised automation system and show the seriousness with which teams are now treating security in these systems.

## Top Support Requests

This next section shows a breakdown of our in-house customer support requests over the last 12 months, by technology.



It’s not surprising to see ActiveMQ, the open source Java-based messaging server, at the top of the list. While powerful, and exceedingly useful, it’s also notoriously under-documented — which makes paid, expert support essential.

The percentage of CentOS support requests is largely tied to customer percentage, as many OpenLogic customers count on us to deliver cost-saving support and security for applications using CentOS and RHEL.

# Today’s World and Looking Ahead

There’s no other way to say it, the state of the world has accelerated our reliance on the software technology that drives human interaction. The need for digital transformation, cloud native infrastructure, and secure, functional code has never been more pronounced. Our gadgets, once novel ways to interact outside of normal social settings, have, seemingly overnight, become our primary means of work, education, and even socialization.

It’s not surprising to see free and/or open software continue to persist and bloom during this transformative time. After all, remote collaborative culture has been the engine of open development since the very beginning. Developers have been using technology to share code and work together on projects for decades.

We’re now seeing in our developer communities a renewed sense of responsibility to the users for whom they develop. Industry interest has begun to focus on verifiability of code, and businesses have begun to think more about security and governance. The conversation is expanding in the mainstream from exploitability to software ethics. How will we, for instance, ensure that the AI we develop over the coming years exists solely to benefit humanity?

Ask the members of OpenAI, whose mission it is to ensure, through open source development and transparent innovation, that it does just that.

With this rise in overall cultural maturity, so have businesses begun to shift their culture. This is perhaps most evident in the early 2020 decision to install Jim Whitehurst of Red Hat as the new President of IBM. As Forbes magazine put it, “Why?  He Gets Culture.” It’s not only the survey results in this important report that demonstrate the truth. You can see it in lobal business behaviors as well. That truth? The market is all-in on free software and has no intention of looking back.

So, what's next, now that a majority of businesses have adopted an open-first policy?  Our survey results would suggest that enterprises are capitalizing on the merger of free software with cloud native infrastructure, and enterprise cloud-native ISVs are circling their respective wagons. When we can pull our software freely from public repositories and deploy our solutions on cloud assets almost as instantly, our barriers to innovation become narrower and narrower. Freely available dependencies work with open  CI/CD technology to perform instant and resilient releases all the time, rather than all at one time.

IBM's purchase of Red Hat, and thereby OpenShift, represented the first major play outside of Google for domination of the container-based cloud native infrastructure market segment.  Open source companies have risen and fallen in short spans of time, clamoring to be part of the "one cloud-native container orchestration solution to rule them all." With SUSE's recent purchase of Rancher, we're in for quite a year as titans and newcomers alike compete on the equal playing field of open innovation.

I hope that the insights you gain from this valuable report will help guide your thinking as the landscape continues to shift under our feet. In the astronomically large and ever-blooming galaxy that open source has become, the more tools we have to navigate, the more effective we will remain in our work, education, and craft.

Justin Reock

Chief Architect, Perforce Software

# Questions?

Have questions about how OpenLogic can support your open source technologies? Get answers from our team of open source experts today.

TALK TO AN OPEN SOURCE EXPERT

# Ready to Build Your Open Source Stack?

Try the OpenLogic Stack Builder. It provides expert advice as you select your open source options. Plus get a free, personalized report based on your choices.

BUILD TECHNOLOGY STACK

# Additional Resources

## Java Resources

- Forrester Report: Weighing the Options for Java Support
- Latest OpenJDK Guide
- Looking for a free OpenJDK Build? Check out our downloads page here.

## Operating System Resources

- Whitepaper: Enterprise Linux at a Fair Price
- Video: Still on RHEL 6? Know Your Options
- Datasheet: CentOS 6 Enterprise Support

## Other Trends in OSS

- A Guide to Open Source for Enterprise
- The Advantages of Ansible Orchestration
- Future of Open Source Software Development