THE STATE OF

Cloud-Native Release Orchestration 2021

circleci

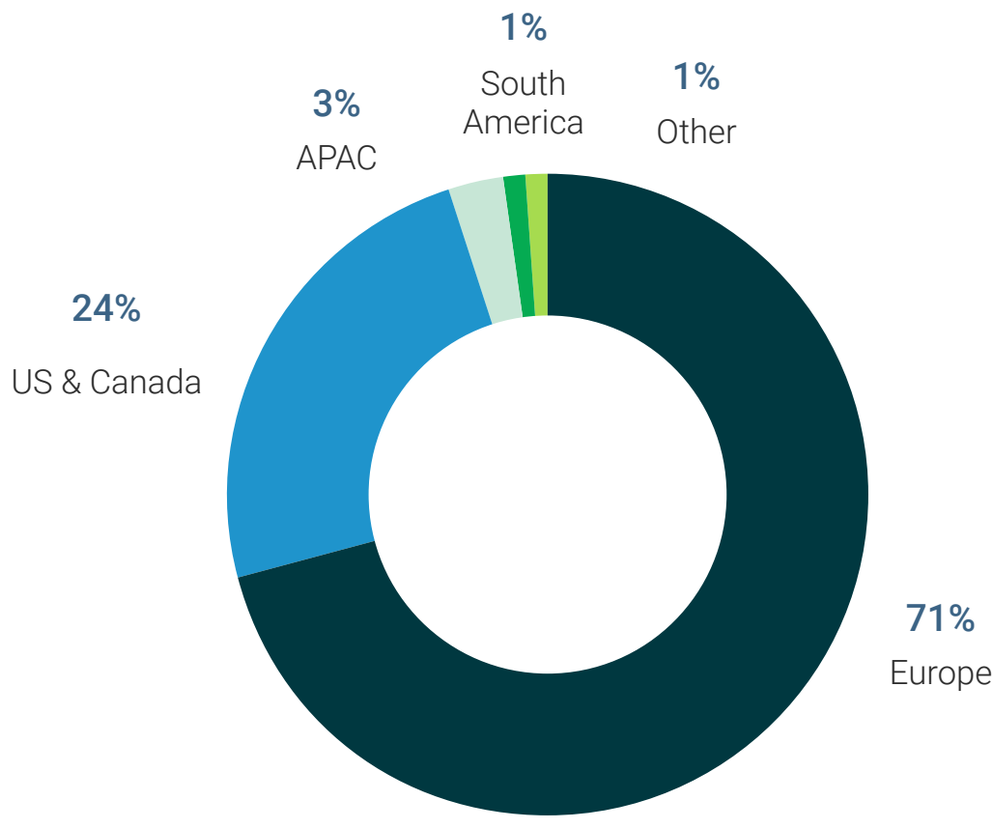
Key Takeaways

- Kubernetes and microservices have become mainstream. 72% uses Kubernetes and 81% uses microservices in production today.
- 86% of participants are still using high-risk release strategies such as rolling update, blue-green and big-bang releases.
- Validation of released software in production remains a manual task. On average it takes 2.2 hours with 4 people engaged per release.
- 53% of rollbacks are done manually with an average time to rollback of 45 minutes.
- Dependency management is the biggest challenge regarding releasing microservices in production.
- Self-service deploy and release automation capabilities are needed to support the shift-right movement of developers.

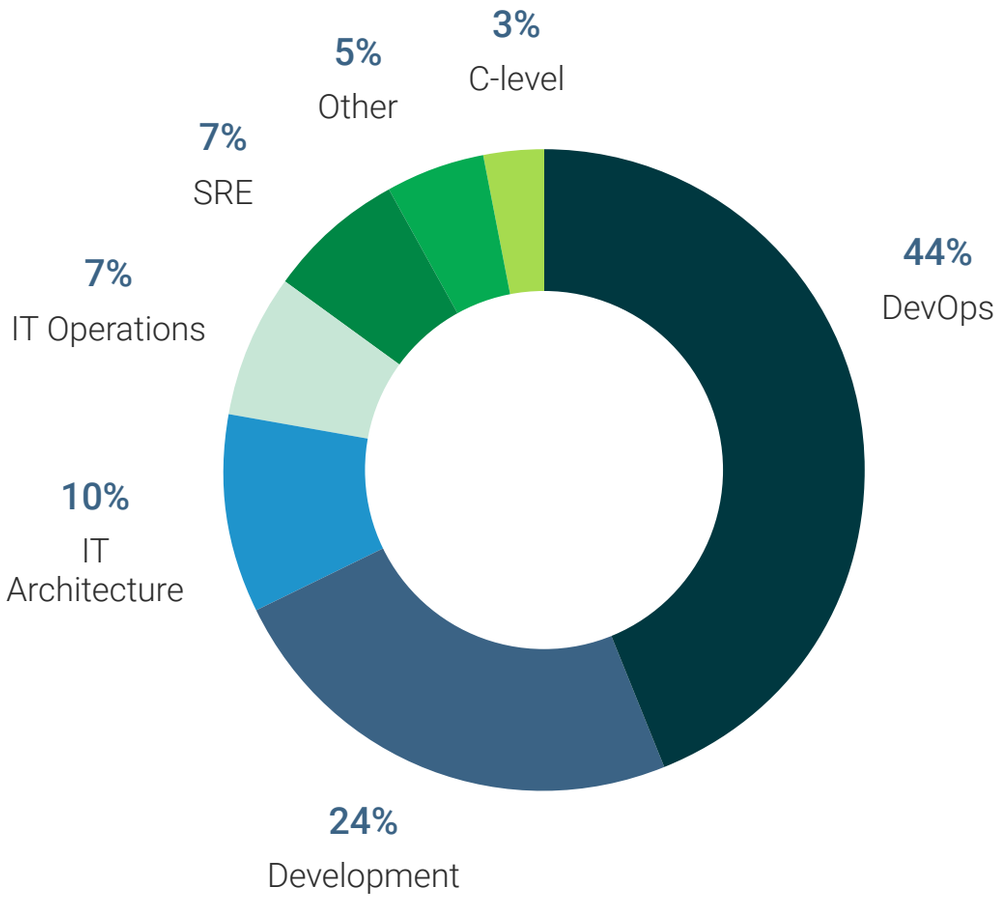
Survey demographics

In total we surveyed 253 developers, DevOps, SREs, engineering managers, architects and C-level executives across the world to uncover trends, challenges and opportunities for improvement when it comes to releasing and validating software in production.

Responses by Region

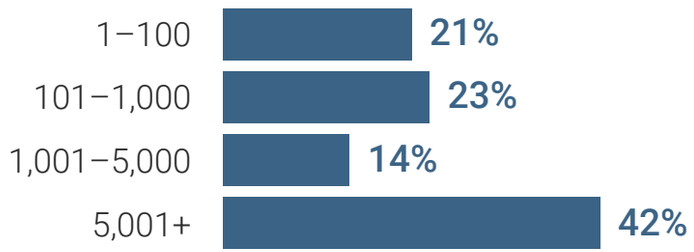


Survey Demographics



Responses by Industry

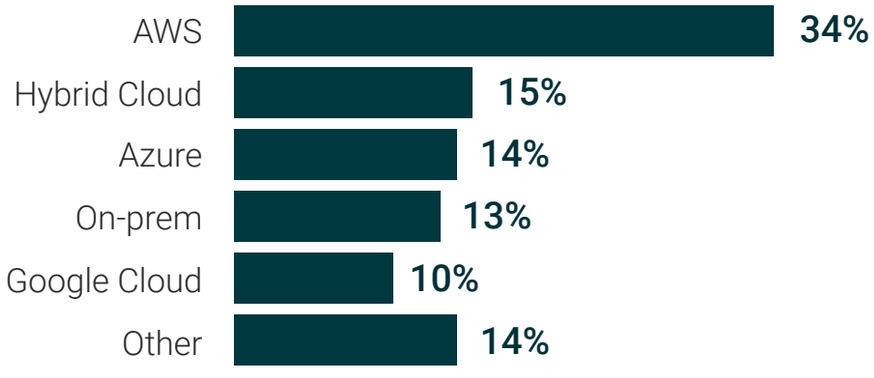
Responses by Company Size



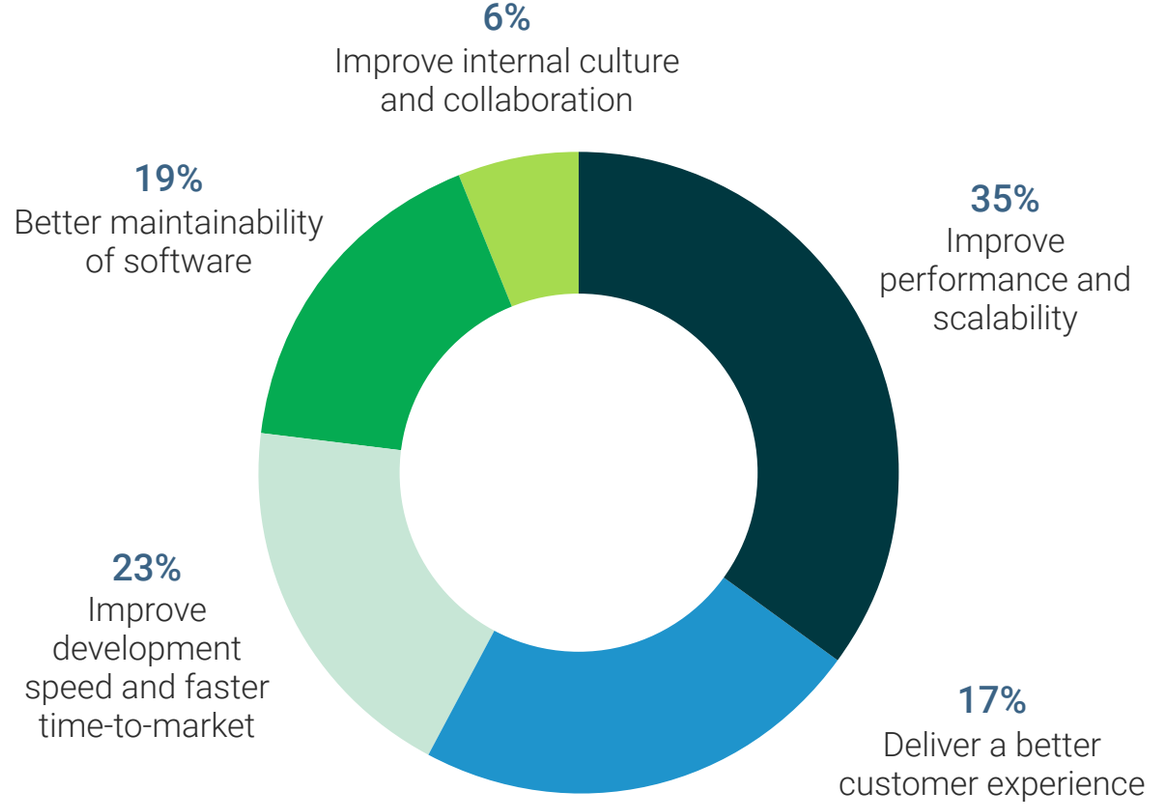
Cloud-Native

Kubernetes and microservices have become mainstream. 72% of participants say that they are using Kubernetes in production today and 16% says that they are evaluating or planning to use Kubernetes. Microservices are even more popular. 81% of participants say that they are using microservices in production today and 14% says they have plans to use microservices.

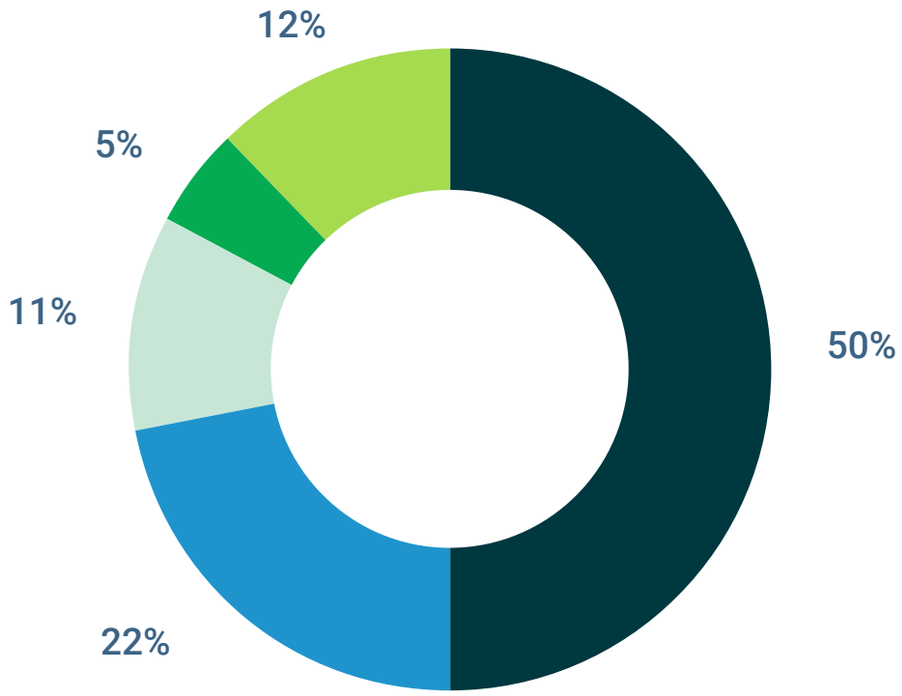
Cloud Providers



Business drivers for Cloud-Native technologies

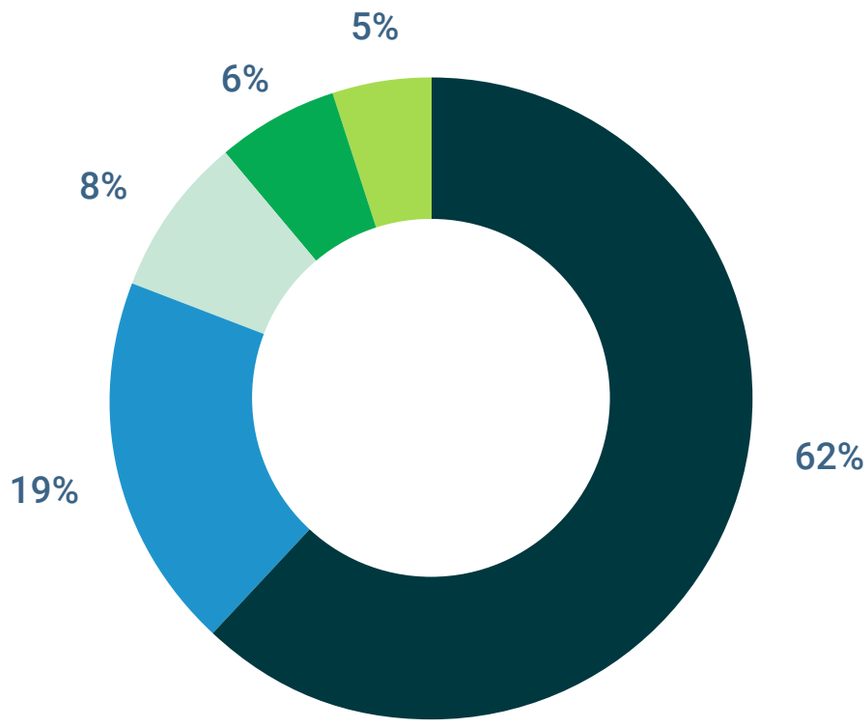


Kubernetes



- We depend on Kubernetes in production today
- We have our first Kubernetes clusters in production
- We are evaluating Kubernetes and it's not in production
- We are planning to implement Kubernetes, but haven't started
- We don't have Kubernetes and have no plans for them

Microservices



- We depend on microservices in production today
- We have our first microservices in production
- We are developing microservices but it's not running in production
- We are planning to develop microservices, but haven't started
- We don't have microservices and have no plans for them

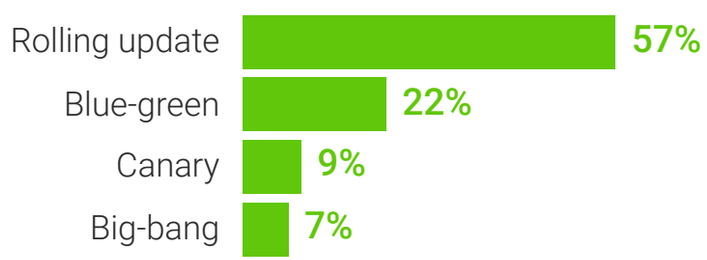
Releasing in production

Releasing in production is the process of taking running code and making it “live” and accessible to users, in a predictable and repeatable way. It’s about paying particular attention to ensure that releasing a new version doesn’t result in a worse experience for your users.

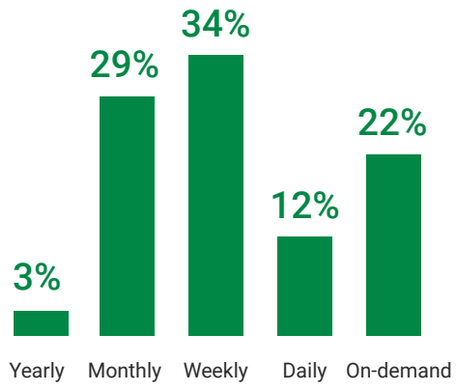
Key insights:

- Majority of participants are still using high-risk release strategies like rolling updates, blue-green and big-bang releases.
- Lack of canary releasing capabilities.
- 68% of participants are releasing on a weekly, daily, or on-demand basis

Release Strategy

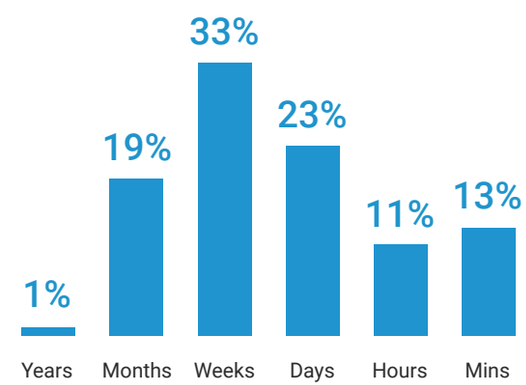


Release Frequency



Companies with a **daily** release frequency have an avg. lead time to production of **hours**

Lead Time to Production



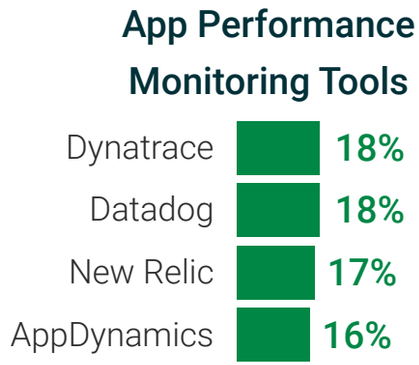
Companies with a **weekly/monthly** release frequency have an avg. lead time to production of **weeks**

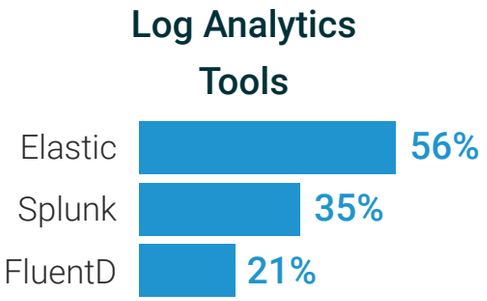
Most used monitoring tools

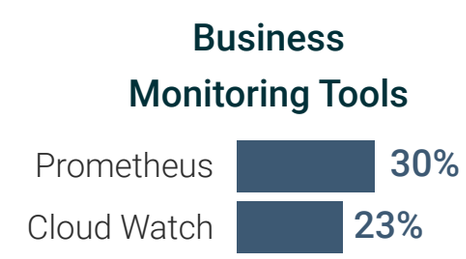
In this survey, we asked what monitoring tools are being used to validate releases and updates in production to ensure health, performance and quality. Please note that participants may use one or more tools.

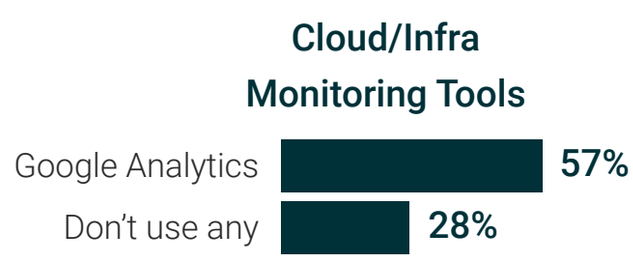
Key insights:

- The data from monitoring tools is not integrated and actionable in the release pipeline.
- Elastic and Prometheus are widely used by participants to validate releases in production.
- Released microservices are often monitored from a technical perspective (response time, latency etc.), but rarely from a business perspective (user experience, orders per second, shopping basket value etc.).









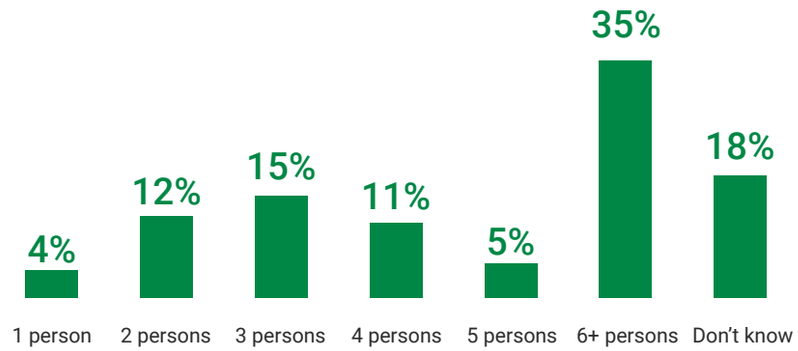
Release validation

Release validation is the process of making sure that released software that's running in production is meeting the requirements in terms of performance, quality, customer experience and business KPIs.

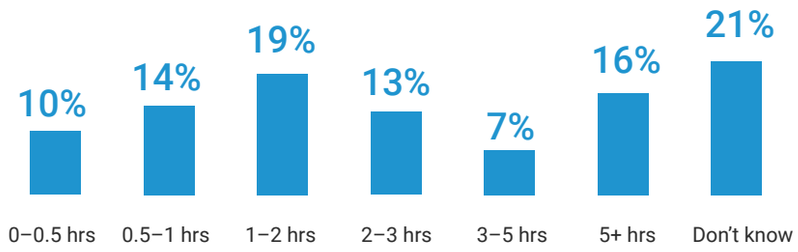
Key insights:

- The validation of released software in production remains a manual task.
- On average it takes 2.2 hours with 4 people engaged during the release validation process.
- Teams that have a higher release frequency are spending less time on validation per release.

Average Amount Of Persons Involved With Release Validation



Average Time Spent On Release Validation

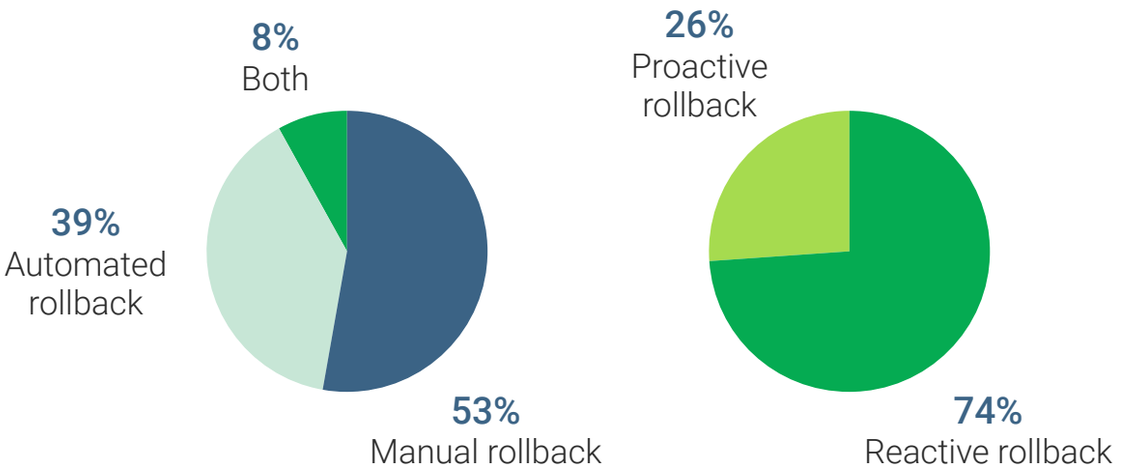


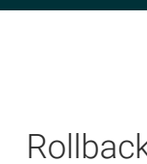
Rollback of releases

Rollbacks are a critical part of your release strategy. When implemented right, it creates a safety net to protect your software teams and minimize the impact on the customer experience when issues are starting to occur.

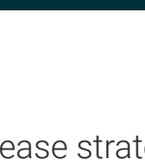
Key insights:

- There is a lack of automation in the rollback process. 53% is still doing manual rollbacks.
- Only 26% of rollbacks are proactive and are used to prevent issues in production.
- Average time to rollback is 45 minutes.



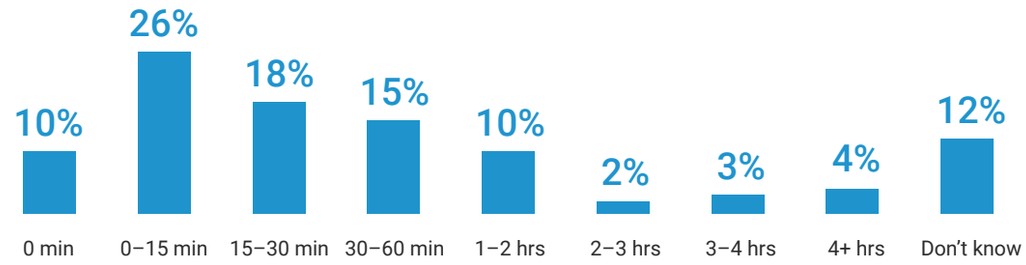


Average time to rollback in a microservices-based environment



Average time to rollback in a monolith-based environment

Time To Rollback After Experiencing Issues



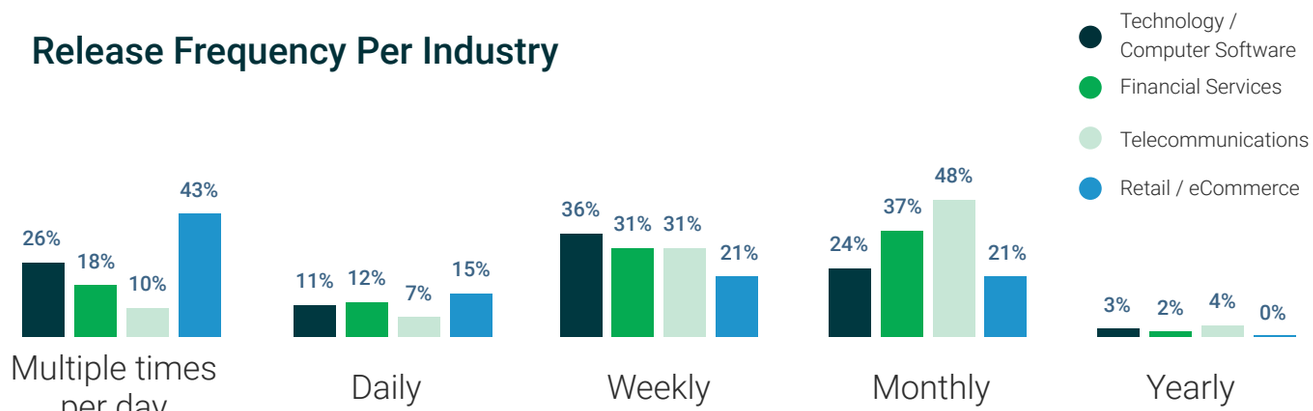
Key metrics per industry and company size

Let's have a look at some of the key release metrics per industry and company size. In this analysis, we've focused on four industries: Computer Software, Financial Services, Telecommunications and e-Commerce.

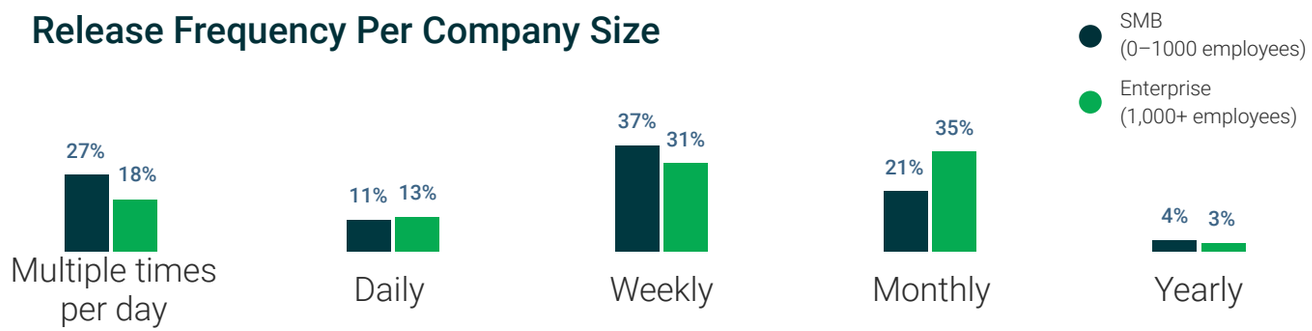
Key insights:

- Retail and e-Commerce companies most often release to production. 43% say they release multiple times per day.
- Small and medium-sized businesses have a slightly higher release frequency compared to enterprise
- 50% of e-Commerce companies say that better maintainability of software is their #1 reason to go cloud-native. Only 7% focuses on improving the customer experience.

Release Frequency Per Industry



Release Frequency Per Company Size

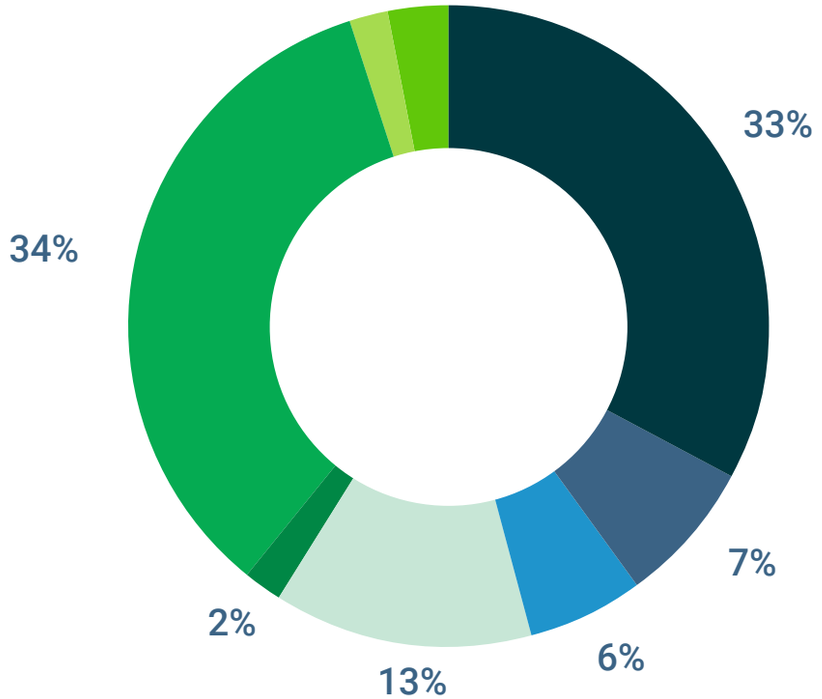


Responsibilities

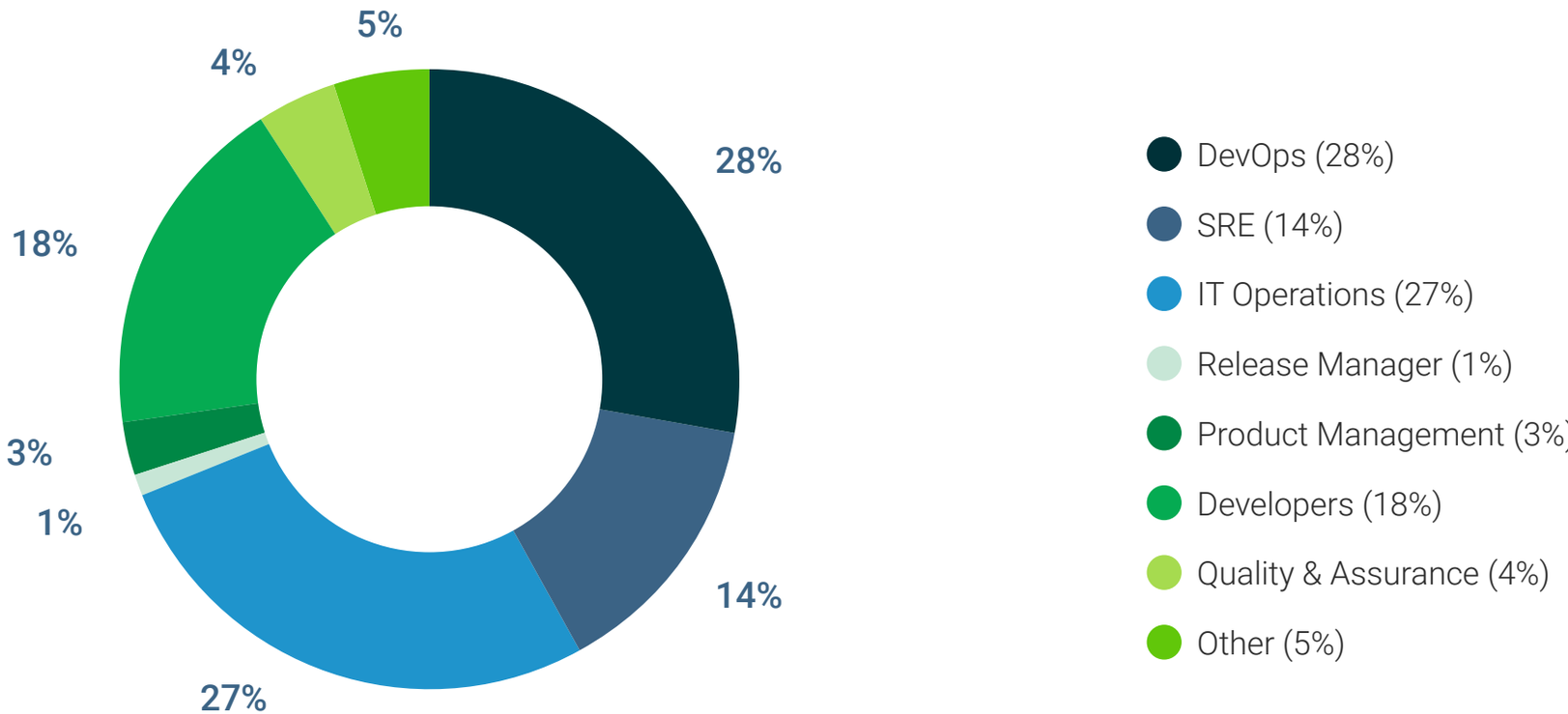
You build it, you run it. Developers are getting more responsibilities and are more engaged with the day-to-day operations of their software. 34% of participants say that their developers are not only responsible for building features, but also for making sure it's released to end-users in production. The question is: how are you enabling your developers in this shift-right approach?

- DevOps (33%)
- SRE (7%)
- IT Operations (6%)
- Release Manager (13%)
- Product Management (2%)
- Developers (34%)
- Quality & Assurance (2%)
- Other (3%)

Who In Your Organization Is Responsible For The Process Of Releasing New Software Into Production?



Who In Your Organization Is Responsible For Monitoring Production Issues And/Or Outages?



Releasing microservices is not without challenges

Here are the top three biggest challenges regarding releasing microservices in production:

- Dependency management — 20%
- Internal processes and communication — 14%
- Validating changes and testing of microservices — 10%

Please note that we've used an open question.

Nearly 9 in 10 report challenges and issues when they release microservices to production

Here's what some of our participants are saying about their biggest challenges regarding releasing microservices in production:

> Validating all services in production still work

> Uncertainty of non-functional behavior especially under load or outages

> Ensuring that they can be indeed released independently

> Difference in data between pre-prod & prod sometimes messes up the release with actual customer data

> Testing the new features and validating the integration with other microservices

> Impact on various chains

> Dependency chain/hell

> Managing microservices: as the number increase managing gets challenging

> To make sure that communication works between microservices

Focus for improvements in 2021

The results are clear. 30% of participants say that they would like to focus on automating the Continuous Delivery and Release processes within their organization. Automation ensures that quality software is released at the speed the business desires and needs to stay ahead.