

A Forrester Consulting  
Thought Leadership Paper  
Commissioned By CloudBees  
June 2021

# Modernizing Software Delivery With End-to-End Automation, Orchestration, And Collaboration



# Table Of Contents

- 3** Executive Summary
- 4** As Dev Footprints Become More Complex, So Does The Toolchain
- 6** The Current State Of Software Delivery Reveals Visibility And Collaboration Deficits
- 10** Business Benefits Scale With Delivery Maturity
- 12** Solve Development Complexity With Automation And Release Orchestration
- 13** Key Recommendations
- 14** Appendix

**Project Director:**

Andrew Magarie,  
Principal Market Impact Consultant

**Contributing Research:**

Forrester's Infrastructure &  
Operations research group

ABOUT FORRESTER CONSULTING

Forrester Consulting provides independent and objective research-based consulting to help leaders succeed in their organizations. Ranging in scope from a short strategy session to custom projects, Forrester's Consulting services connect you directly with research analysts who apply expert insight to your specific business challenges. For more information, visit [forrester.com/consulting](https://forrester.com/consulting).

© 2021, Forrester Research, Inc. All rights reserved. Unauthorized reproduction is strictly prohibited. Information is based on best available resources. Opinions reflect judgment at the time and are subject to change. Forrester®, Technographics®, Forrester Wave, RoleView, TechRadar, and Total Economic Impact are trademarks of Forrester Research, Inc. All other trademarks are the property of their respective companies. For additional information, go to [forrester.com](https://forrester.com). [E-49428]



Delivery automation, paired with an organizational structure that enables collaboration and communication, drives top- and bottom-line success for the business.

## Executive Summary

Increasingly complex software delivery is stifling many delivery organizations' potential and endangering their DevOps transformation success. Growing development ecosystems and toolchains, paired with legacy organizational structures, lead to visibility and collaboration challenges that impact key business objectives and can even lead to serious security and risk concerns. To solve these challenges, firms are adopting software delivery automation capabilities and solution orchestration platforms to improve visibility across the DevOps toolchain. Delivery automation, paired with an organizational structure that enables collaboration and communication, drives top- and bottom-line success for the business.

CloudBees commissioned Forrester Consulting to evaluate current software delivery maturity. Forrester conducted an online survey with 317 IT decision-makers responsible for software development lifecycle (SDLC) investment decisions at US and European enterprises to explore this topic.

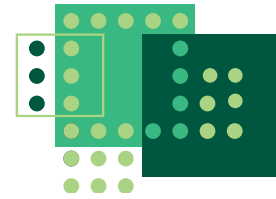
### KEY FINDINGS

- › **Common workflows across software delivery improve visibility, collaboration, and consistency.** Common workflows are an organizational building block of software delivery maturity that drive consistency between teams and processes through improved collaboration and visibility. Creating common workflows across the software delivery organization is an important step toward software delivery maturity — and is the largest organizational maturity gap in our study.
- › **Unifying tools enables better visibility and collaboration across teams and process.** Mature software delivery organizations enable developers to use best-of-breed solutions, but doing so can create complexity that impedes visibility, collaboration, and delivery processes. Firms can solve complexity challenges by unifying the delivery toolchain through solutions that provide integrated information across the SDLC in one location, enabling better visibility, management, measurement, and audit of software delivery activities across teams and processes.
- › **Higher software delivery maturity unlocks continuous business benefits.** Organizations that pair delivery automation capabilities with effective collaboration practices report more successful digital transformations and exceed business expectations at a higher rate than those that cannot. Software delivery maturity is correlated with improved market share, user adoption, innovation, and customer loyalty. These benefits also scale with maturity, meaning there is value to improvement at any stage of the journey.

# As Dev Footprints Become More Complex, So Does The Toolchain

Developers and DevOps teams today must support portfolios that span both cloud and on-premises applications across an expanding number of platforms. Most businesses have cloud-native and traditional data center strategies for modern and heritage application deployments. To accommodate this hybrid model, many software organizations have adopted new technologies, using multiple tools to support development capabilities across the software development lifecycle. Our survey of US and European SDLC investment decision-makers shows:

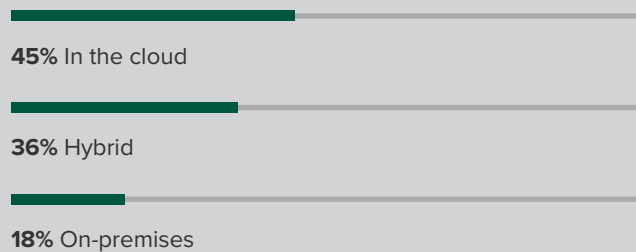
- › **Organizations are moving to cloud-native, but hybrid and on-premises still make up most of the portfolio.** Just under half (45%) of survey respondents say most of their portfolio runs in the cloud, meaning on-premises development still plays a significant role in the enterprise for most businesses (see Figure 1).
- › **The emergence of new platforms to support is increasing the development footprint.** The accelerating shift to the cloud makes platforms like container (43%) and serverless (38%) the future for software development. However, virtual machines are still the most prevalent cloud development platform (70%), and mainframe development is still a priority for four in 10 decision-makers. In fact, three in four say their organizations are supporting multiple platforms today. This means software delivery organizations must manage multiple generations of applications, often spanning multiple groups and creating inherent complexity in the delivery organization.



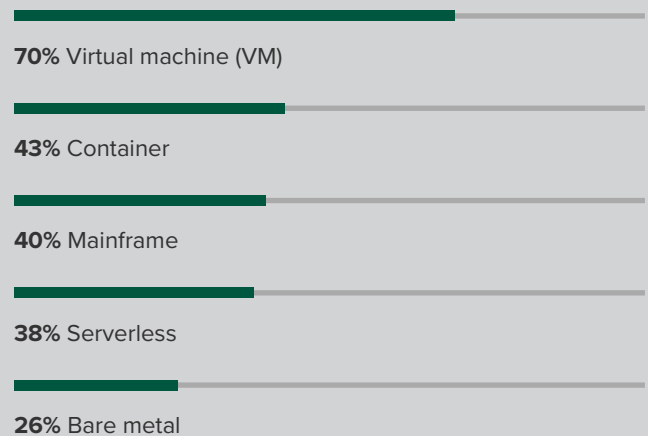
Three in four respondents say their firms are supporting multiple development platforms today.

Figure 1

“Where does the majority of your organization’s application portfolio run?”\*



“For which of the following target platforms is your software development organization developing software, whether in the cloud or on-premises?”



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises

\*Note: Percentages do not total 100 because of rounding.

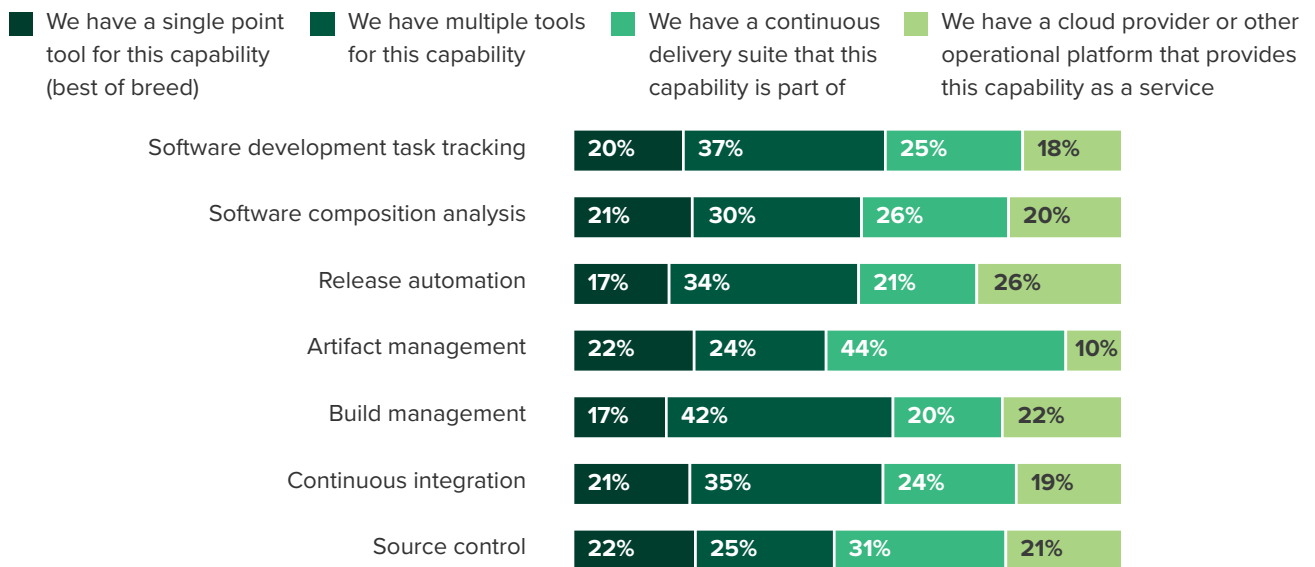
Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

- › **Many software delivery organizations adopt complex tooling to accommodate increasingly diverse varieties of target platforms.** To support multiple development types and platforms, many organizations are adopting multiple tools to provide the same delivery capabilities (see Figure 2). While this is inherently not a problem, multiple tools across groups or platforms can create complexity that slows down delivery, makes it more difficult to understand delivery status, and creates other visibility and collaboration issues.
- › **Other organizations try to lock down the toolchain, creating additional challenges.** Close to four in 10 decision-makers are going the other way, creating required standardization around a single set of tools and solutions. This decision can eliminate some of the complexity issues of having multiple tools but often means that a single delivery solution suite is not optimized for every one of the development platforms it must support, creating additional inefficiencies and challenges.

A single delivery solution suite is often not optimized for every one of the development platforms it must support, creating additional inefficiencies and challenges.

Figure 2

“How does your software development organization support the following development capabilities?”



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises

Note: Percentages may not total 100 because of rounding.

Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

# The Current State Of Software Delivery Reveals Visibility And Collaboration Deficits

The complexity of the delivery ecosystem is increasing. Effective software delivery practices require organizational visibility, communication, and a fit-for-purpose, simplified toolchain. Firms that excel at these capabilities have more successful DevOps transformations, exceed business objectives, and grow revenue faster than those that do not.

To determine how mature firms' software delivery capabilities are today, we assessed survey respondents on software delivery automation and organizational capabilities (see Figure 3). We then grouped respondents into high (top 25%), medium (middle 50%), and low (bottom 25%) maturity groups, based on their current capabilities. By looking at similarities and differences between maturity groups, we can uncover strengths and weaknesses for different levels of maturity.

**Figure 3**

*Statements evaluated on a 1-to-5 scale*

Software delivery automation	Software development organization
1. We manage and version all deployable assets using an artifact repository, container registry, or similar.	1. We have clear and accepted channels/protocols for interteam communication across the SDLC that are followed.
2. All development teams are able to push code all the way to production.	2. We have strong interteam coordination to execute on shared goals across the SDLC.
3. I have all the information I need in a single report to understand current software release status.	3. Affected stakeholders have visibility into other teams in the SDLC's initiatives that may have an impact on them.
4. I have all the information I need to understand where the bottlenecks are in the SDLC process to improve them.	4. We have common workflows and process concepts across all teams for software delivery.
5. We practice continuous deployment with automation from commit through testing, build/integration, package, to final release.	5. Software team members feel they have an important role to play in the organization's results.
6. I have all the information I need to understand the <b>business</b> outcomes of our software releases.	6. We routinely pass all security and regulatory compliance audits and tests.
7. I have all the information I need to understand the <b>technical</b> outcomes of our software releases.	7. We are able to easily integrate new software delivery services (e.g., adding monitoring to production or code scanning to continuous integration) and provide these services to various teams in a self-service way.
8. We use modern software delivery capabilities (process and tools) across the portfolio, including all technology stacks.	8. Developers in our organization run the systems that they build.
9. Our entire software portfolio is using feature flags.	

Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

## ORGANIZATIONAL CAPABILITIES

When we discuss organizational capabilities, we are focusing on how software teams work together and communicate to deliver high-quality software releases. Communication and effective collaboration, along with common workflows, services, and processes, are critical to delivery success. However, even the best organizations face challenges. When analyzing organizational capabilities across the maturity groups, we found:

- › **Visibility and coordination challenges impede all levels of maturity.** As DevOps initiatives scale up, coordination and visibility become increasingly important; without them, broader business objectives suffer, and teams start to thrash. Overall, respondents scored lowest on interteam coordination and visibility across the SDLC (see Figure 4). Fewer than 40% of high-maturity respondents report strong coordination and high visibility across the SDLC, with the respective percentage of low-maturity respondents *in the single digits*.
- › **Integration of new capabilities is difficult for most.** The current trend is for dedicated platform teams to run software delivery services (source, CI/CD, testing, etc.). The longstanding challenge for shared services is timely and responsive delivery. Fewer than half of high-maturity respondents we surveyed can easily integrate new delivery capabilities and provide them to the organization as self-service. Here again, challenges multiply for lower-maturity respondents.

Fewer than 40% of high maturity respondents report strong coordination and high visibility across the SDLC.

Figure 4

“How much do you agree with the following statements about your software development organization?”  
(Showing “Strongly agree”)



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises  
Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

- › **In addition, lower-maturity groups struggle with common workflows and processes.** While too much process can stifle an organization, it's important to have common ways of working. Having common processes and workflows provides a basis for teams to collaborate and communicate in the same context, improving alignment and visibility between teams and engendering better predictability across processes in the SDLC. This allows an organization to focus more on delivering value. The largest gap in capabilities between high-maturity organizations and others involves common workflows and processes across all delivery teams.

## DELIVERY AUTOMATION

Software delivery automation capabilities encompass the technology capabilities required to effectively push software from development to production, as well as the information needed to understand requirements, bottlenecks, and outcomes of software release. While some maturity groups have the technology foundations of software delivery automation in place, all groups could improve their software delivery automation visibility capabilities. Analyzing software delivery automation capabilities across maturity groups, we found:

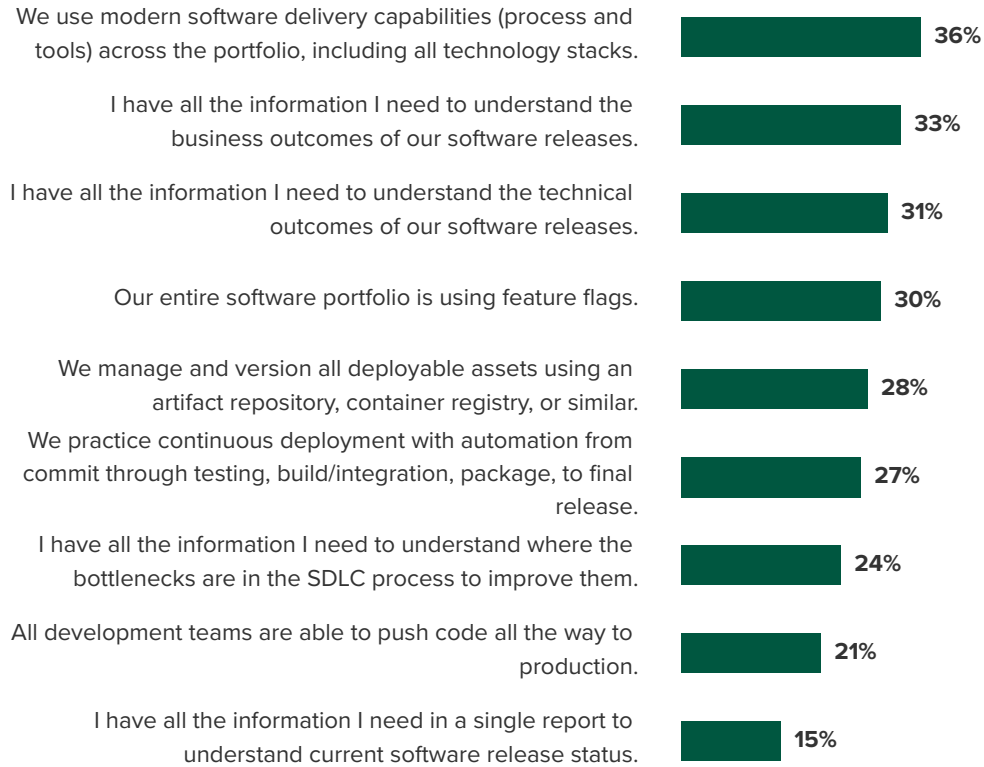
- › **The top challenges involve the understanding of key aspects of the delivery pipeline.** As above, coordination is an increasing challenge as organizations scale to teams of teams. Once again, visibility is a major shortcoming for organizations of all maturity levels (see Figure 5). Fewer than three in 10 high-maturity respondents say they have all the information they need in a single report to understand software release status, dipping to a woeful 2% for low maturity. In addition, fewer than half of high-maturity respondents, and only one in 10 low-maturity, can pinpoint the bottlenecks in their delivery processes in order to fix them.
- › **Few organizations enable all development teams to push code all the way through production.** True continuous deployment is a worthwhile idea as it pushes organizations to new heights of systems maturity and automation. Only one-third of high-maturity decision-makers say their firms allow all development teams to push code through production.
- › **Low-maturity organizations lack the key technology underpinnings of software delivery automation.** Leading decision-makers recognize package and artifact management as essential control points; vulnerabilities and technical debt are best managed here. Feature flags enable continuous deployment by decoupling deployment from user-level release. While 17% of low-maturity respondents say they are using modern delivery capabilities across their software portfolio, just 5% manage and version deployable assets and only 11% use feature flags.



**Figure 5**

**“How much do you agree with the following statements about your software delivery automation (SDA) capabilities?”**

(Showing “Strongly agree”)



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises  
Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

# Business Benefits Scale With Delivery Maturity

Organizations that combine key software delivery automation capabilities with strong organizational visibility and collaboration are better positioned to meet key business goals and realize successful DevOps transformations. The benefits of mature software delivery practices tend to scale, meaning there is value to improvement at any stage of transformation. Organizations that score higher on our maturity model are more likely to:

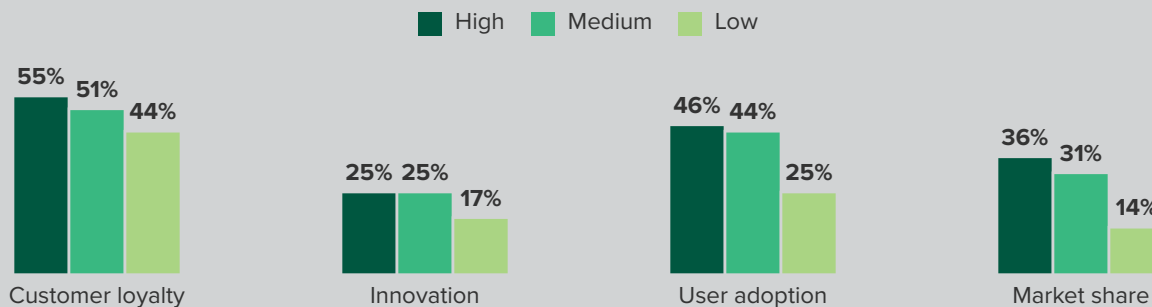
- › **Succeed with their DevOps transformation.** High-maturity respondents in our survey are more than five times as likely than low-maturity respondents to say that their DevOps transformation is exceeding business expectations (35% vs. 6%).
- › **Grow revenue faster.** High-maturity organizations are more likely to grow revenue at a higher rate than their less mature peers — almost three times as likely to be growing at 15% or more year over year.
- › **Exceed business expectations for key goals.** More mature delivery organizations tend to outperform their peers across a wide range of business metrics (see Figure 6). Maturity is highly correlated with business objectives like improving market share, user adoption, innovation, and customer loyalty. Organizations that have the toolchain flexibility and effective collaboration practices to foster innovation can delight users and customers alike, which is a competitive differentiator in the market. The greatest jump in incremental value occurs from low maturity to medium, meaning it is essential to enable effective automation, visibility, and collaboration practices across the delivery organization.
- › **Adapt to changing market conditions.** More mature delivery organizations have greater flexibility to adapt to changing market conditions, such as the COVID-19 pandemic. High-maturity respondents in our survey are nearly twice as likely than low-mature respondents to say their organizations responded to the pandemic better than their peers (49% vs. 27%).

Software delivery maturity is highly correlated with business objectives like improving market share, user adoption, innovation, and customer loyalty.

Figure 6

## Business Outcomes Exceeding Expectations

(Showing “Exceeded (0% to 5%) expectations” and “Significantly (>5%) exceeded expectations”)



Base: 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises  
Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

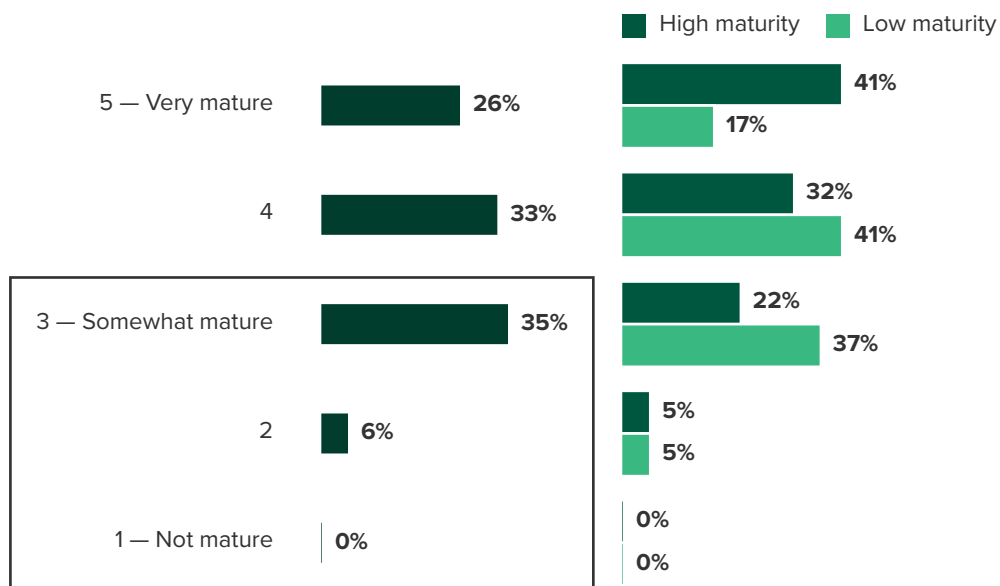
## AUTOMATION, VISIBILITY, AND COLLABORATION IMPROVE GRC CAPABILITIES

Survey respondents identify security as the most important priority for software delivery solutions — but also the most challenging priority to meet. In addition, just 25% rate their governance, risk, and compliance (GRC) capabilities for software delivery as very mature, with 40% saying they are not mature (see Figure 7).

Recent data breaches highlight the dangers of a disconnected delivery toolchain. Additionally, a lack of visibility and communication across the organization can be a catalyst for compliance risk and increased security threats. Stronger visibility and collaboration capabilities across teams can reduce potential GRC issues in software delivery. And 41% of high-maturity respondents rate their GRC capabilities as very mature, compared to just 17% of low maturity.

Figure 7

“On a scale of 1 to 5, how mature are your governance, risk, and compliance capabilities for software delivery in terms of people, process, and technology?”



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises

Note: Percentages may not total 100 because of rounding.

Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

# Solve Development Complexity With Automation And Release Orchestration

As the software landscape and ecosystem both increase in complexity, many organizations end up adding additional DevOps tools to address new needs and preferences. An ever-expanding portfolio of diverse applications and technology stacks, rife with complex application dependencies and workflows, inevitably leads to visibility, collaboration, and security challenges that can limit even advanced delivery organizations.

Smart decision-makers know that they need to balance SDLC stakeholders' freedom of choice with concerns for standardization and security. Modern software delivery automation processes should be applied across the entire portfolio, enabling comprehensive visibility and collaboration. To improve software delivery maturity and reap the benefits of a high-performing organization, SDLC decision-makers should:

- › **Apply modern software delivery automation processes across the portfolio.** Automation is a key enabler for software delivery, as it allows developers to focus on their key creative tasks while ensuring that quality checks and other release imperatives are routinely covered.
- › **Reduce complexity while still enabling teams to select best-of-breed solutions where they have a business case.** High-maturity respondents are more likely to say their organizations use best-of-breed solutions across the SDLC, while low-maturity decision-makers say their firms are more likely to use a continuous delivery suite. To enable groups to use best-fit solutions, but mitigate visibility and complexity issues that can arise, organizations should look for solutions that help integrate and orchestrate their toolchains. This solution should provide integrated information across the SDLC in one location, enabling better visibility, management, measurement, and audit of software delivery activities.
- › **Adopt platform teams with standardized communication and collaboration channels.** Forrester's research shows that a product-centric build/run model is correlated with exceeding business expectations. Product teams incorporate stakeholders from across the SDLC, better aligning teams to overall business goals and enabling better communication and visibility.<sup>1</sup>

To enable groups to use best-fit solutions, but mitigate visibility and complexity issues that can arise, organizations should look for solutions that help integrate and orchestrate their toolchains.

# Key Recommendations

Forrester's in-depth survey of SDLC decision-makers about software delivery maturity yielded several important recommendations:



**Automate routine SDLC tasks, such as testing, build/package, and deployment.** Firms should automate tests should be automated, and don't forget quality assurance like static analysis and software composition analysis. This is table stakes, but many organizations are still on the journey.



**Unify continuous delivery and release automation (CDRA) tools into one platform and manage them as a product.** Product-centric operating models are a major new trend: Everything is a product. Whether external- or internal-facing, there is much to learn from customer-focused product management. The customer here, of course, is the development team, whose satisfaction should be assessed on a regular basis.



**Publish and provide access to platforms (such as CDRA) in an enterprise service portal.** What good is a product you can't find? Make discovery and onboarding seamless and efficient. This is why service catalogs are enjoying a resurgence, especially in the era of remote work.



**Keep the platform open to best-of-breed solutions where justified.** Technology is complex, and there may be good reasons for point solutions for specialized problems. You may have diverse platforms: mainframes, midranges, etc. Or perhaps you are applying continuous delivery techniques to a database. In any case, plan for integrations to achieve maximum value.



**Implement value stream management.** Value stream management is key to illuminating how well your pipeline is flowing. If you have data in a common repository, leverage it to illustrate blockages to value. If you have point solutions, ensure their data is integrated into the common platform.



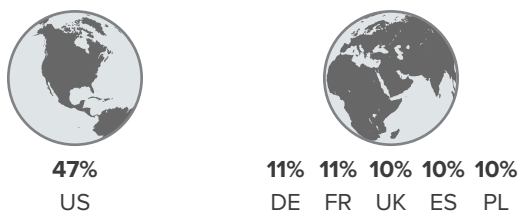
**Leverage package management as a key control point in the software delivery supply chain.** It's all about what you're running in production. Package management is where it all comes together: the things you've built, the things you've downloaded, and the things you've purchased. It's increasingly a virtual concept, with well-known artifact managers, container registries, OS package managers, and even application stores all vying for prominence in managing the all-important binary.

# Appendix A: Methodology

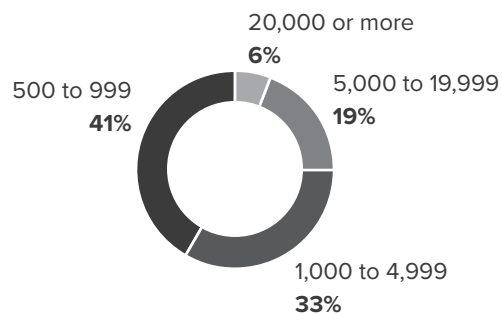
In this study, Forrester conducted an online survey of 317 respondents in France, Germany, Poland, Spain, the UK, and the US to evaluate the current state of software delivery maturity. Survey participants included IT and application development decision-makers responsible investment decisions across or for a portion of the software development lifecycle. The study was completed in January 2021.

# Appendix B: Demographics

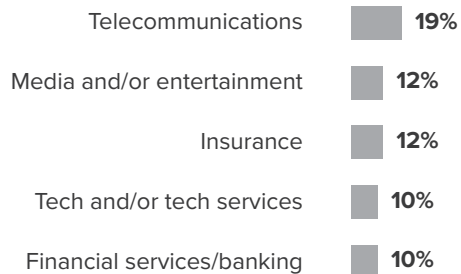
## REGION/COUNTRY



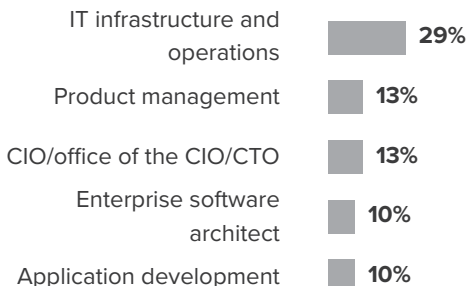
## COMPANY SIZE (EMPLOYEES)



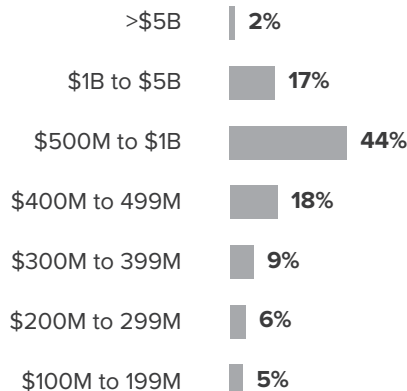
## INDUSTRY (TOP 5 SHOWN)



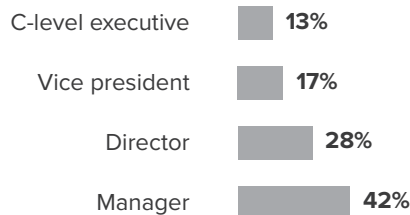
## FUNCTION (TOP 5 SHOWN)



## COMPANY REVENUE



## RESPONDENT LEVEL



Base: : 317 IT decision-makers responsible for SDLC investment decisions at US and European enterprises

Note: Percentages may not total 100 because of rounding.

Source: A commissioned study conducted by Forrester Consulting on behalf of CloudBees, January 2021

## Appendix C: Supplemental Material

### RELATED FORRESTER RESEARCH

“The New Digital Operating Model Calls For Product Teams; The EPOCH Model Will Help You Get There,” Forrester Research, Inc., November 5, 2020.

“Your Operating Model Must Enable Innovation,” Forrester Research, Inc., November 5, 2020.

“The Future Of Technology Operations,” Forrester Research, Inc., August 28, 2020.

## Appendix D: Endnotes

<sup>1</sup> Source: “The New Digital Operating Model Calls For Product Teams; The EPOCH Model Will Help You Get There,” Forrester Research, Inc., November 5, 2020.