

Deeper Than Digital

Top-performing modern enterprises show why
more perfect software is fundamental to success



Table of Contents

Introduction: What is the best way to manage the software life cycle?	03
1. Cloud Maturity	09
2. Observability Mastery	11
3. End-to-End Data Visibility	13
4. Digital Resilience	15
5. Free-to-Thrive Teams	17
Conclusion	19

Introduction:

What is the best way to manage the software life cycle?

In the past decade, managing and developing software has evolved from being solely a function of IT to a set of capabilities that are central to business competitiveness and survival. In today's world of digital transformation, the most successful businesses are mastering software development, performance optimization, and life cycle management.

But how can organizations ensure that their approach to developing and managing the software life cycle will put them ahead of the competition?

The latest research from New Relic, based on a survey of 400 senior business leaders across six countries, has uncovered an advanced group of organizations that demonstrate excellence across five critical pillars of the software life cycle. We call this group the "More Perfect Software Leaders."

The data shows that these leaders are not only outperforming other firms when it comes to the software life cycle, but they are also reporting better performance across a range of metrics, including financial. This suggests that achieving excellence across the five pillars could translate into higher overall performance.

In this report, we explore in detail what sets the More Perfect Software Leaders apart from the 25% of the research sample that have made the least progress across all of the five pillars—a group that we call the "More Perfect Software Laggards."

What is "more perfect software"?

Contrary to what you might think, more perfect software is not software that does not fail. In our survey questionnaire, we defined it as "software that is dynamic and resilient enough to deliver outstanding customer experiences."

More perfect software

More perfect software and systems are developed and maintained in a way that makes it easy to quickly understand how end users are interacting with them. They also allow for greater agility, because developers can quickly identify the root cause of any issues and make rapid updates and iterations. More perfect software is developed to be more perfect from inception. The result? It can withstand a degree of “failure” without causing expensive catastrophes.

That resilience gives developers the confidence they need to innovate at speed.

More perfect software encompasses many factors, but we believe there are five imperatives for companies striving to build and maintain it. These are the five pillars that formed the basis of our research.

1. Cloud Maturity

Running the majority of applications and infrastructure in the cloud so that agility and speed to market are improved

2. Observability Mastery

Having a single view of how software and systems are performing in real time, allowing for holistic insights that can shape software design and development, and enabling teams to rapidly identify and resolve issues

3. End-to-End Data Visibility

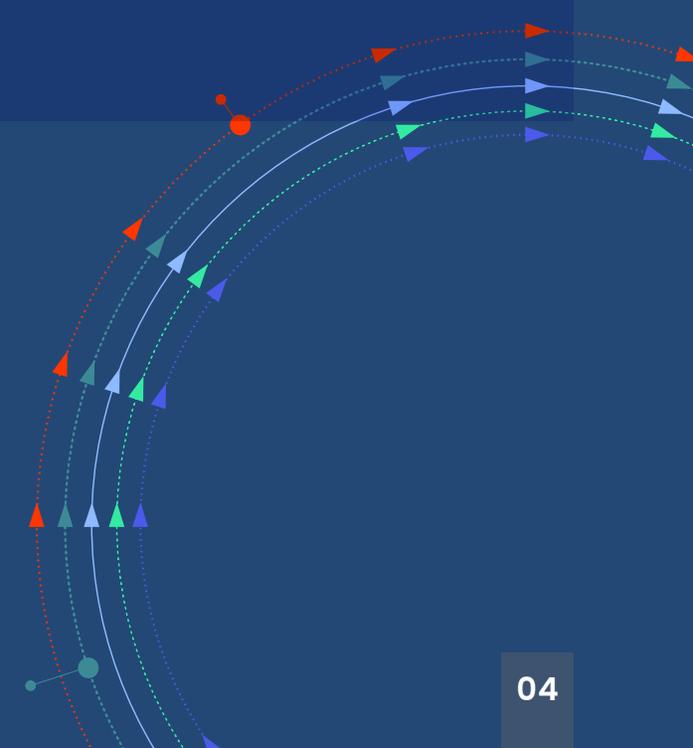
Integrating software performance data with customer experience and business data to enable companies to develop solutions that are precisely aligned to customers’ needs

4. Digital Resilience

“Building for failure,” and employing chaos testing and automated remediation to reduce errors and downtime while enabling faster innovation

5. Free-to-Thrive Teams

Empowering teams with the authority, data, and tools to make better and faster decisions that keep customers happy (and enable the business to succeed)



How much progress have firms made in the journey to more perfect software?

To measure the extent to which they have achieved mastery across the five pillars of more perfect software, we scored organizations on a scale from 0 to 10 (with 10 being best), based on their performance in each pillar (see “Research methodology” at the end of this report). The scores across all five pillars were then aggregated and averaged to give an overall score that acts as a proxy for software development and management excellence.

The research finds that firms are only about halfway on their journey to developing and managing outstanding software: The average score is 6.22 out of a possible 10. However, the More Perfect Software Leaders are significantly further ahead, with an average score of 7.57. The More Perfect Software Laggards have an average score of 4.39.

Japan has a significantly higher proportion of firms in the leader group (24%) than the other countries, which are Australia (10%), US (14%), UK (15%), France (17%), and Germany (19%).

To find out your organization’s score and see how it compares with those of the leaders and laggards, use the [More Perfect Software benchmarking tool](#).



BRAND PERCEPTION



EMPLOYEE
ENGAGEMENT



FINANCIAL
PERFORMANCE



PACE OF INNOVATION

Who are the More Perfect Software Leaders?

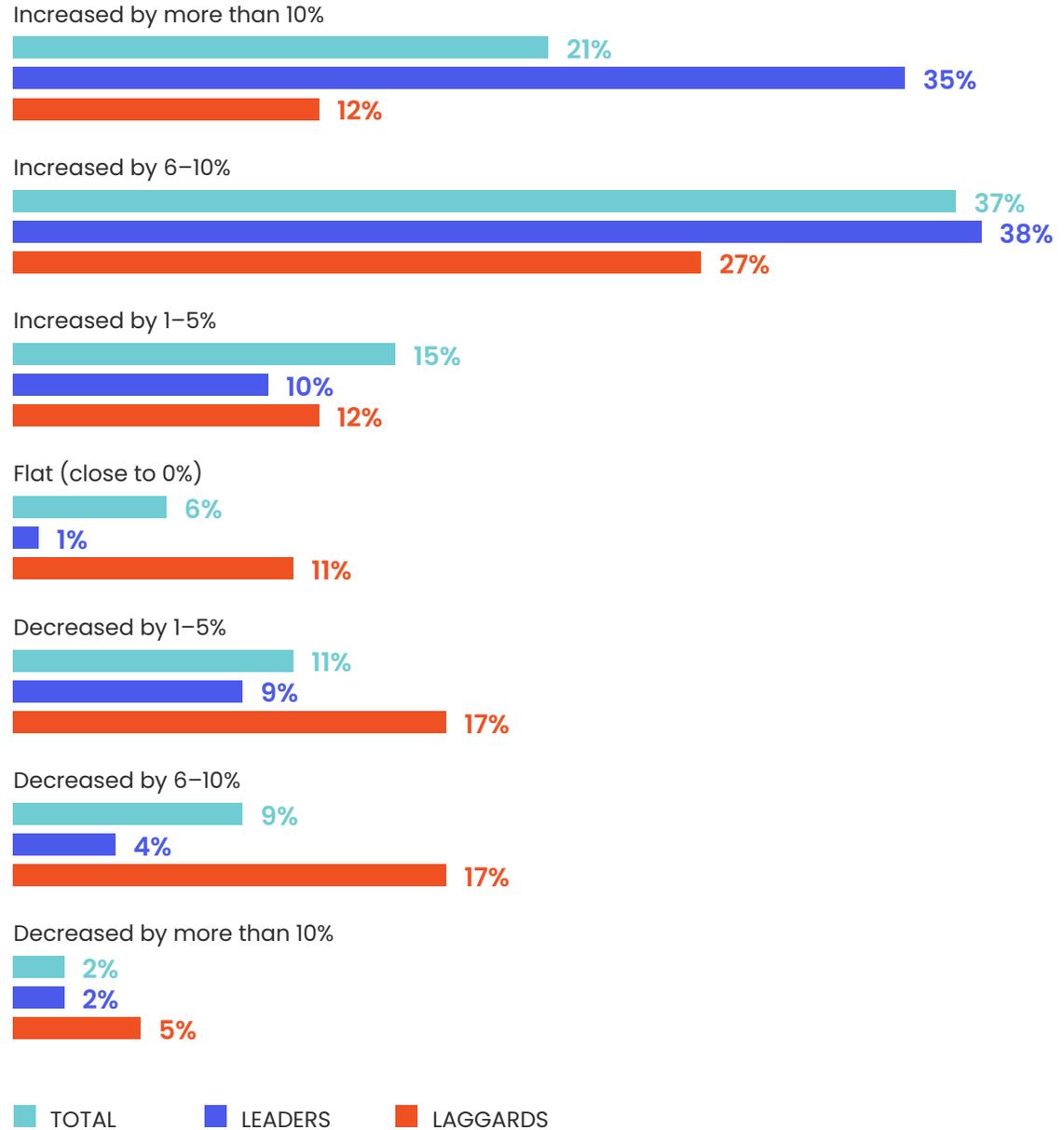
The leaders are a group of firms—about 25% of the sample—that report excellent progress across all five pillars of software life cycle management.

They believe that more perfect software matters: 90% say that developing and deploying more perfect software is a strategic priority for their organization.

Leaders report better performance than the rest of the sample across a number of key areas. Not only do they report higher revenue growth in the past three years, they are also more likely to say that their firm is ahead of the competition in the following areas:

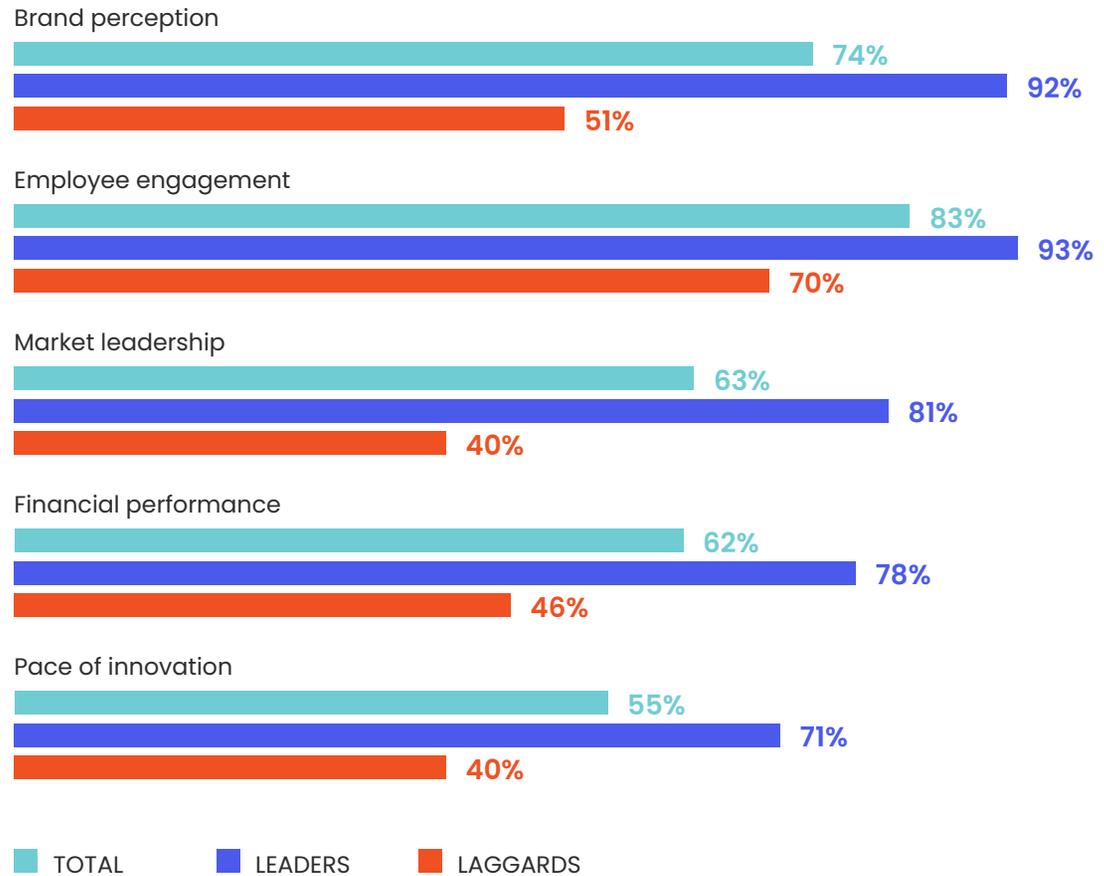
- Brand perception
- Employee engagement
- Financial performance
- Pace of innovation

Leaders show stronger revenue growth in the past three years



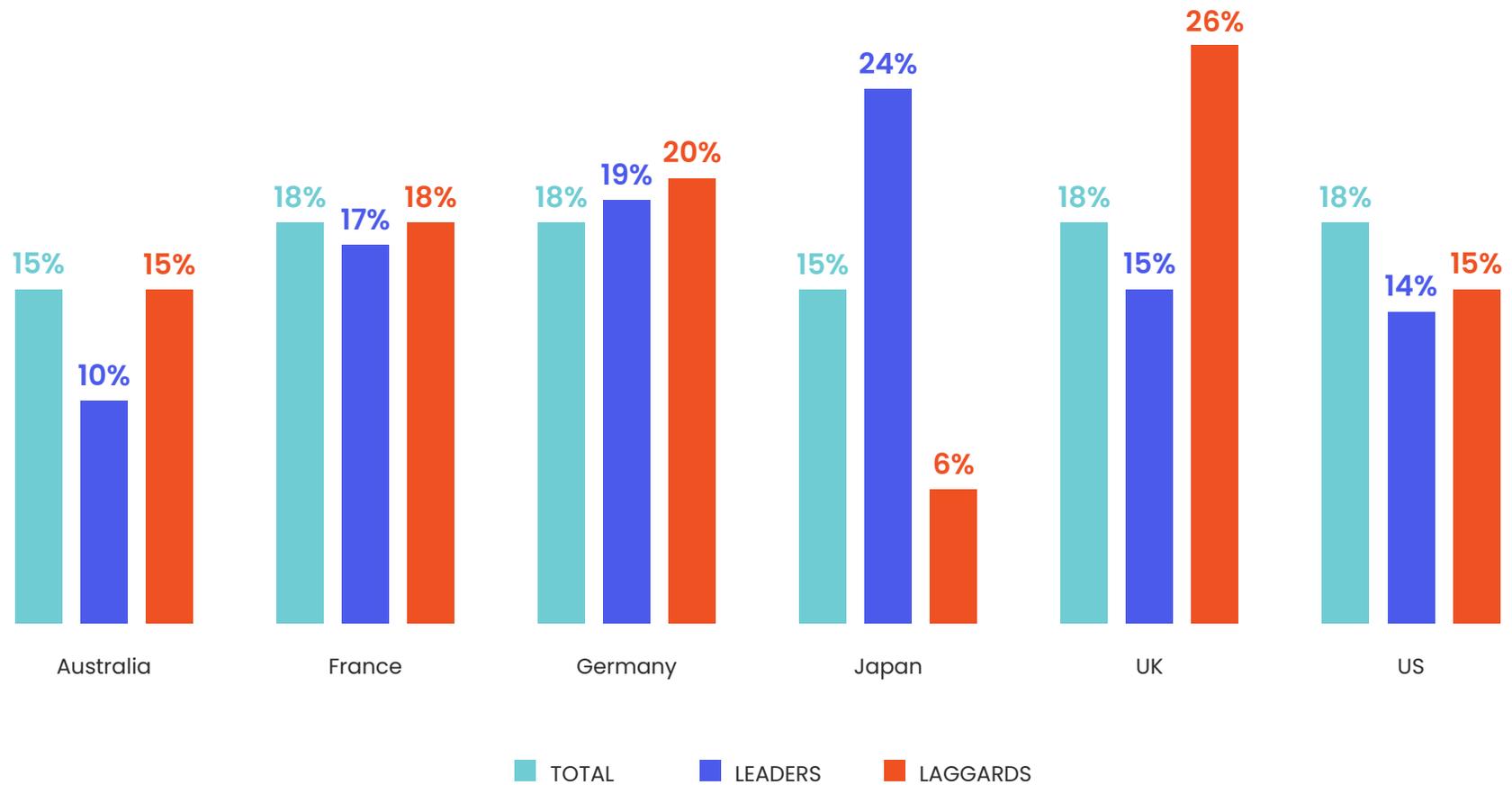
Q5: How has your organization's annual revenue changed over the past three years?
Base: total = 400; leaders = 99, laggards = 94

Leaders are more likely to report that they are ahead of their competitors on brand perception, employee engagement, and market leadership (ahead/far ahead)



Q6: How does your organization compare with its competitors in the following areas?
Base: total = 400; leaders = 99, laggards = 94

Japan has a higher proportion of More Perfect Software Leaders



Q1: Which country is your main place of work?
Base: total = 400; leaders = 99, laggards = 94

1. Cloud Maturity

Here, we dive into the specific activities and traits that make the leader group's approach to software development and management so successful, and explain what sets them apart from the laggards.

Migrating systems to and developing applications in the cloud is vital for producing, deploying, and managing software at a rapid pace.

On average, the More Perfect Software Leaders run 66% of their software and systems in the cloud, while for the More Perfect Software Laggards it is only 46%. And leaders are significantly more likely than laggards to have adopted a cloud native approach to software development (96% compared with 24%) and to CI/CD (98% compared with 40%).

The research also reveals that the barriers to greater cloud adoption differ between the leaders and the laggards. For the leaders, the top obstacles are security concerns and regulations; for the laggards, lack of skills and sunk investment in on-premises infrastructure are the top two barriers.



96%

of leaders have adopted a cloud native approach to software development

Compared with

24%

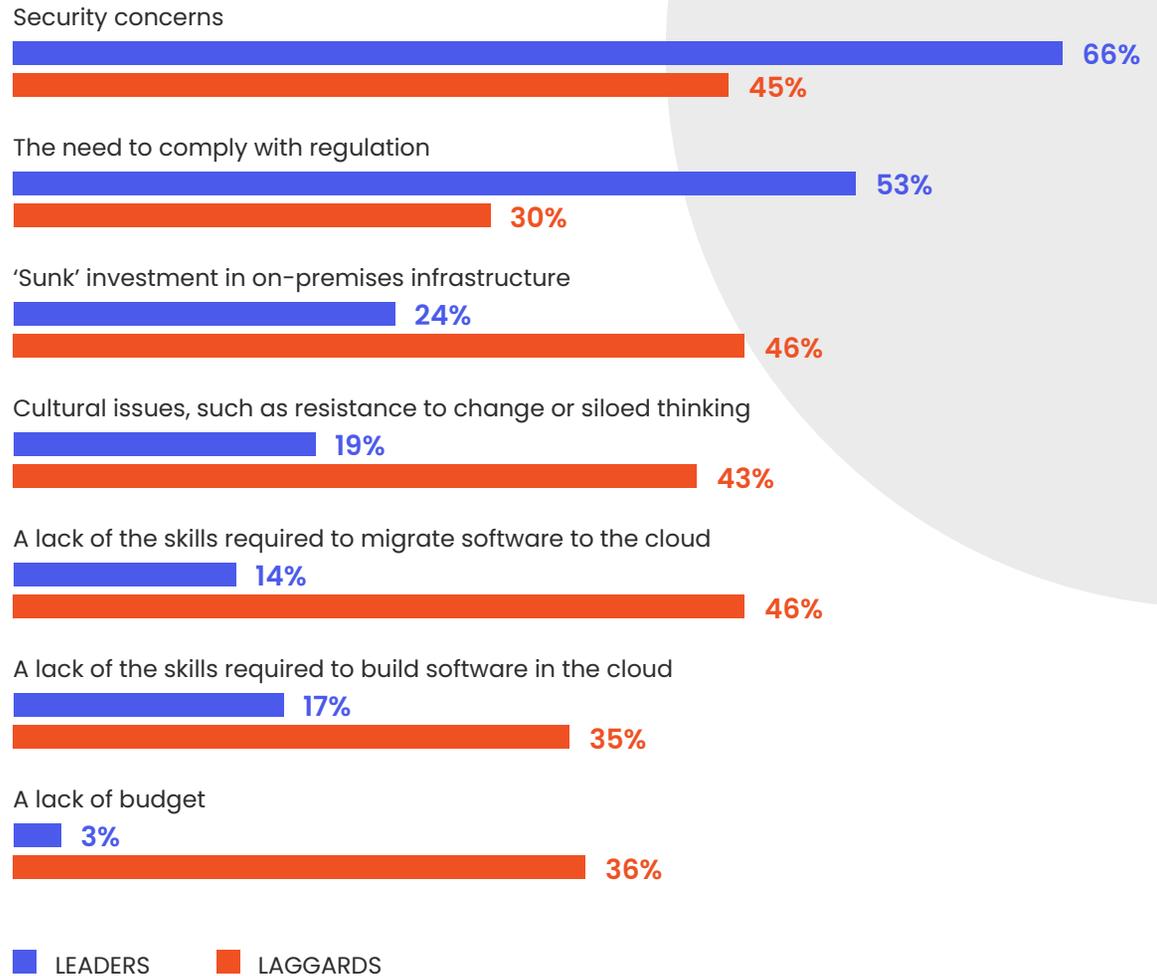
of laggards

“Developing our cloud native capabilities is the No. 1 priority for our IT team today.”

Sreesh Inguva

CTO, Cloud and Digital Transformation,
Fujitsu America

Barriers to cloud migration



Q10: What are the top three barriers to your organization's journey to the cloud?
Base: total = 400; leaders = 99, laggards = 94

2. Observability Mastery

94%

of leaders say observability is key to developing software



Compared with

56%

of laggards

Our research defines “observability” as automatically monitoring in real time all software and infrastructure performance data—metrics, events, logs, and traces—to rapidly deliver a holistic understanding of an IT stack’s operation and performance.

Observability is a critical component of software development and performance optimization, because it allows developers to rapidly innovate, test, and deploy new features and applications with the confidence that it won’t break other parts of the system. It also helps to reduce downtime by enabling site reliability engineering (SRE) and DevOps teams to rapidly diagnose, troubleshoot, and resolve issues and understand not just what happened, but why it happened.

The More Perfect Software Leaders know the importance of observability: 94% say that this is key to developing software—not just something bolted on afterwards. Just 56% of the laggards say the same.

Leaders also report higher levels of observability than laggards. The vast majority (78%) find out about system and software interruptions via observability platforms, compared with only 12% of laggards. Their use of observability platforms is one probable reason why leaders are less likely than laggards to report that they learn of software and service interruptions via external customers (33% compared with 48%).

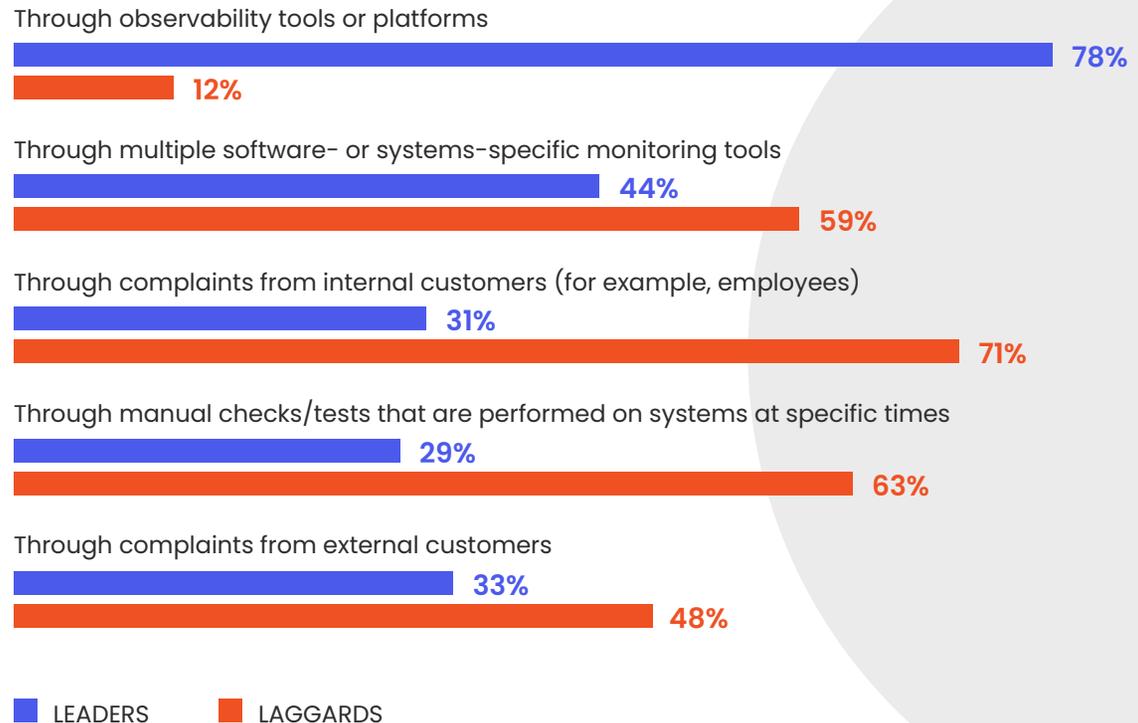
The data also shows that a lack of observability is a major barrier to improving business performance: 79% of laggards agree that this is the case, compared with only 12% of leaders.

“Observability means being able to make changes and not have to worry about breaking things. If you don’t have confidence that you can make a change without affecting a bunch of different things, you’re not going to be making those changes, and therefore you’re not going to be innovating.”

Leo Guinan

DevOps Engineer, Fuse by Cardinal Health

How IT teams learn about software and service interruptions



Q12: How does your organization’s IT team usually learn about software and system interruptions?
Base: total = 400; leaders = 99, laggards = 94

3. End-to-End Data Visibility

“Observability doesn’t mean anything without context. We need to decide what we want to observe and what kind of insight we want to reach. When you have the goal you can make a plan to reach that goal, but the goal needs to be clear.”

Marco Consolaro

Co-author of *Agile Technical Practices Distilled: A Journey Toward Mastering Software Design* and coach at Cazoo

Another factor that sets apart the More Perfect Software Leaders from the More Perfect Software Laggards is their ability to integrate data on software and systems performance with data on end-user performance (across platforms). All leader organizations have done this, compared with just two in 10 laggard organizations.

Firms that integrate data in this way can quickly understand how changes to software affect their customers and overall business performance. Their developers can create an experimental loop by first understanding the impact of a change, and then focusing their innovation where it is most likely to deliver results. It also allows developers to actively infer where issues are and why they exist so they can be fixed more quickly. Almost all of the leaders (99%) say they are able to quickly understand the results of changes to software, compared with only 38% of laggards.

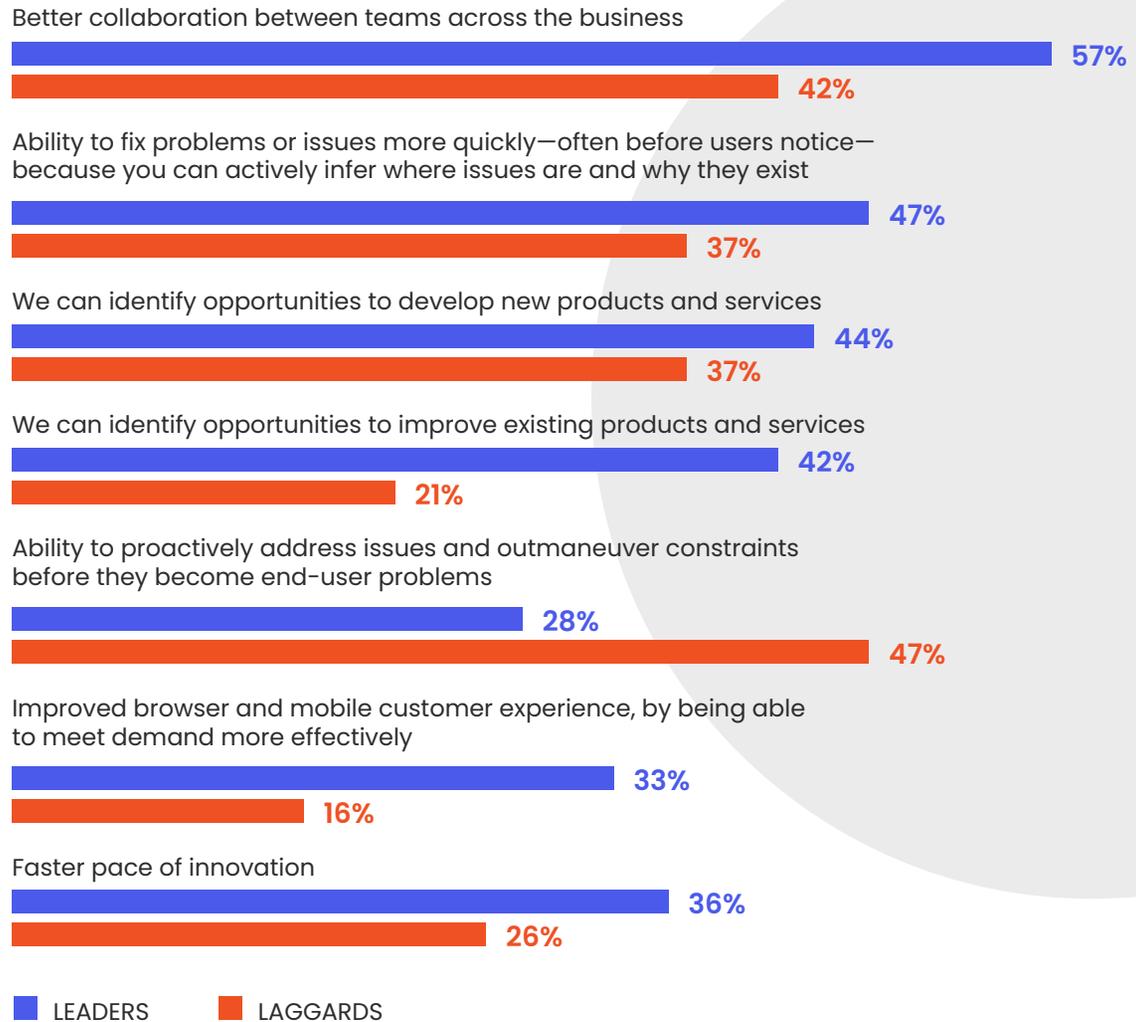


Benefits seen as a result of integrating backend software and systems performance data with end-user web browser and mobile performance data

“We want to know what the user is experiencing when they are on our site. This is particularly important for us, as there are a number of different third-party integrations that are happening within a user’s browser, which have the potential to slow things down. The New Relic tool monitors these integrations and shows us what the end impact is to our consumers.”

David Michael

Director, Consumer Platform Operations,
Meredith Corporation



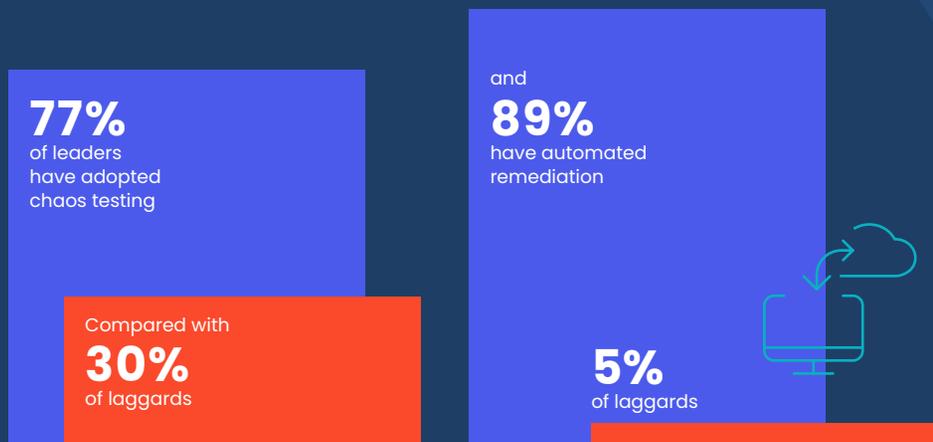
Q18b: What benefits has your organization seen as a result of integrating its backend software and systems performance data with its end-user web browser and mobile performance data?
Base: total = 283; leaders = 99, laggards = 19

4. Digital Resilience

Almost every firm tries to minimize software failures, but top-performing firms “build for failure.” Primarily, this means employing approaches such as chaos testing and automated remediation to dramatically improve architecture resilience, which gives teams the confidence to make rapid changes in response to changing customer needs.

The More Perfect Software Leaders are far more likely than the More Perfect Software Laggards to have adopted chaos testing (77% compared with 30%) and automated remediation (89% compared with 5%).

The leaders’ focus on resilience appears to be paying off. The majority of this group (83%) have fewer than five major software or systems outages per month, compared with just 3% of laggards. Issue resolution times are also faster for the leaders: 75% boast an average resolution time of less than 30 minutes, compared with only 1% of laggards.

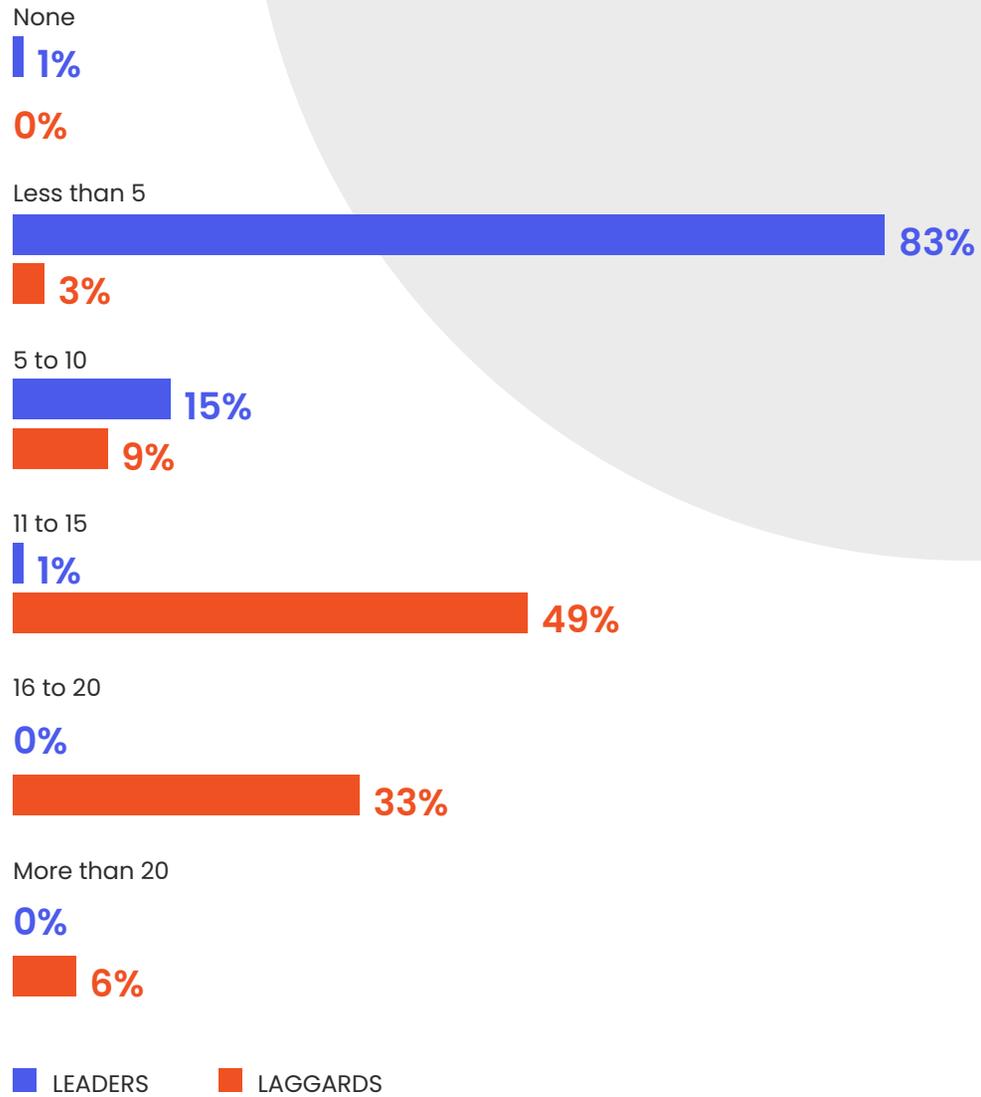


“Reducing the amount of time it takes to diagnose and fix problems gives us more capacity for innovation.”

Leo Guinan

Software Engineer, Cardinal Health

Major software and systems outages per month



Q21: Approximately how many major software or systems failures/outages does your organization experience each month? Base: total = 400; leaders = 99, laggards = 94

5. Free-to-Thrive Teams

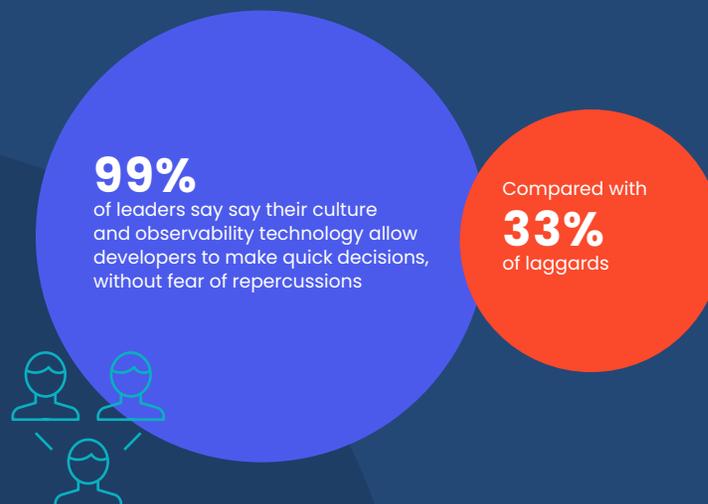
“I don’t think that we could have moved forward with this large re-platforming effort if we didn’t have the approval and willingness of our senior leadership.”

David Michael

Director of Consumer Platform Operations,
Meredith Corporation

Our final pillar of more perfect software focuses on how development and operations teams work. To produce better software more quickly, firms must give developers the autonomy to make decisions about software development and management. Too often, an overly hierarchical team structure and a complex maze of legacy systems mean that developers spend too much time navigating internal politics and fixing issues, and not enough on innovation and development.

The More Perfect Software Leaders (99%) are more likely than the More Perfect Software Laggards (33%) to say they have in place the observability technology and culture to allow developers to make quick decisions, without fear of repercussions. This might be why leaders spend on average 77% of their time innovating (versus fixing issues), while laggards spend just 54%.



“Our culture and observability technology allow our developers to make quick decisions, without fear of repercussions.”

“A key factor in developing better software is teams’ ability to work independently. Can organizations achieve the Amazon ideal of the ‘two pizza team,’ where each team can independently develop, test, and deploy value to customers? Or do they have 40 different, interconnected teams that must always be communicating, coordinating, prioritizing, synchronizing, and scheduling together? If it’s the latter, then nothing can get done.”

Gene Kim

DevOps researcher and WSJ bestselling author

Leaders



Laggards



■ AGREE ■ STRONGLY AGREE

Q24a: Thinking about your organization’s software developers and how they work, to what extent do you agree or disagree with the following statements?
Base: total = 400; leaders = 99, laggards = 94

Conclusion

Three key messages emerge from the research:

1. The five pillars of more perfect software define the operational foundation of digital excellence.

They underpin more dynamic, sophisticated, and effective software development, management, and operation. Organizations across sectors can benefit by strengthening their capabilities across each of the five pillars.

2. Excellence across the pillars is associated with high-performing businesses.

The More Perfect Software Leaders are more likely than the More Perfect Software Laggards to think they are ahead of their peers in a range of business areas, including brand perception, employee engagement, market leadership, financial performance, and pace of innovation (see page 7).

3. The five pillars will only get more important in the 2020s.

This is due to three factors: The first is growth in customer demand for premium functionality and customizability (thanks to the expectations set by the likes of Google and Amazon). The second is the increase in software development complexity (e.g., the integration of more datasets, more AI, and more emerging technologies). The third factor is greater collaboration and integration with third parties (e.g., greater cross-platform integration and agility) as industries mature their digital capabilities.

“I love **Jon Smart’s definition of DevOps**: better value, sooner, safer, and happier. It means that organizations can innovate more quickly and rapidly deliver value to their customers while preserving reliability, stability, and security. In my view, organizations that master this new mode of production will be the ones that survive and win.”

Gene Kim

RESEARCH METHODOLOGY

The insights in this report are based on a survey of 400 business leaders conducted in February and March 2020. The respondents are based in six countries: Australia (60), France (70), Germany (70), Japan (60), the UK (70), and the US (70). All respondents are employed in organizations with annual revenues of at least \$500 million, and 20% of respondents work within the C-suite.

The More Perfect Software Leaders are the 25% of respondents that scored highest across 12 key questions within the survey (see at right), and the laggards are the 25% of respondents that scored lowest across these questions.

Cloud Maturity

- What percentage of your organization's software and systems run in the cloud (private or public), with the remainder kept on-premises?
- To what extent has your organization adopted a "cloud native" approach to software development?

Observability Mastery

- How would you rate your organization's ability to understand why software and systems are not working (e.g., the speed and ease with which you can find and diagnose issues)?
- What percentage of your organization's IT systems are instrumented?
- What is your average time to detection (MTTD) for software issues?

End-to-End Data Visibility

- To what extent has your organization integrated its data on software and systems performance with its data on end-user browser and mobile performance (e.g., the customer experience) to understand how each affects the other?
- To what extent can your organization query all performance monitoring data, connect data from third-party systems, and build custom dashboards that represent how all software and systems are performing and interacting?

Digital Resilience

- To what extent has your organization adopted chaos testing/engineering?
- To what extent has your organization adopted automated remediation?
- To what extent has your organization adopted continuous integration and continuous delivery?

Free-to-Thrive Teams

- Thinking about your organization's software developers and how they work, to what extent do you agree or disagree with the following statements?
- Our dev teams can quickly understand the result of changes to software (e.g., active inference), so a rapid experimental loop is in place.
- Our culture and observability technology allow our developers to make quick decisions, without fear of repercussion.

