RESEARCH REPORT

Optimizing CI/CD Pipelines

A quantitative study of the financial impact of software failures in CI/CD pipelines

2020

Executive Summary

What is the real cost of persistent failures in Continuous Integration (CI) pipelines?

That is the question which a recent survey conducted by a Cambridge Judge Business School MBA project sought to answer. The research team leveraged a two-fold approach to attain quantifiable data. They conducted extensive secondary research across industry databases and analyst firms, and deployed a field survey to gather primary research data.

The research reached three key findings:

1. **Adoption of CI best practices is on the rise**
   88% of enterprise software companies say they have adopted CI practices, compared to 70% in 2015

2. **Reproducing software failures is impeding delivery speed**
   41% of respondents say getting the bug to reproduce is the biggest barrier to finding and fixing bugs faster; and 56% say they could release software 1–2 days faster if reproducing failures wasn't an issue

3. **Failing tests cost the enterprise software market $61 billion**
   This equals 620 million developer hours a year wasted on debugging software failures

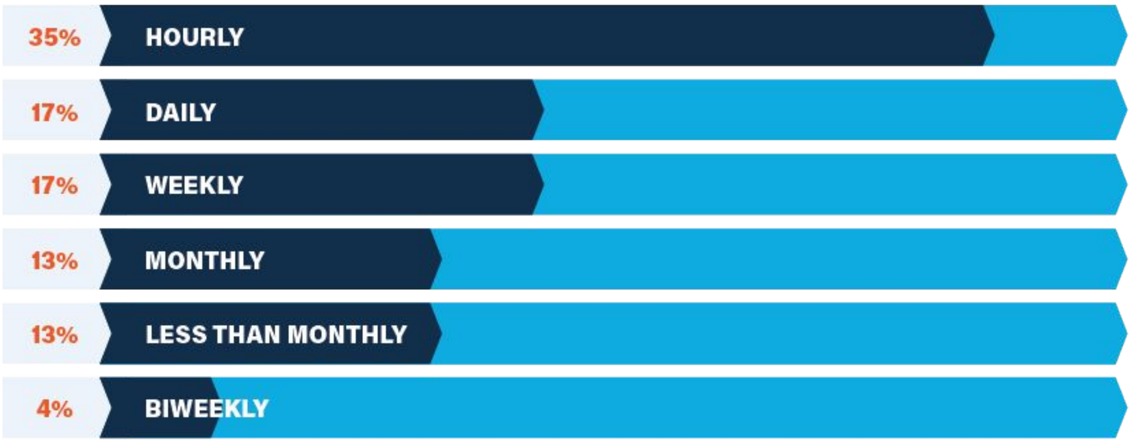
Every company is a software company. The ability to **deliver high-quality software at speed** has become a primary source of competitive advantage. Yet the growing backlog of software failures in test remains a major impediment to delivery speed. Why? Because developers cannot easily reproduce the defects, so it takes weeks (or months) to debug before the software can be shipped. *Reproducibility* is the fundamental technical problem that slows engineering teams down.

Key Finding 1

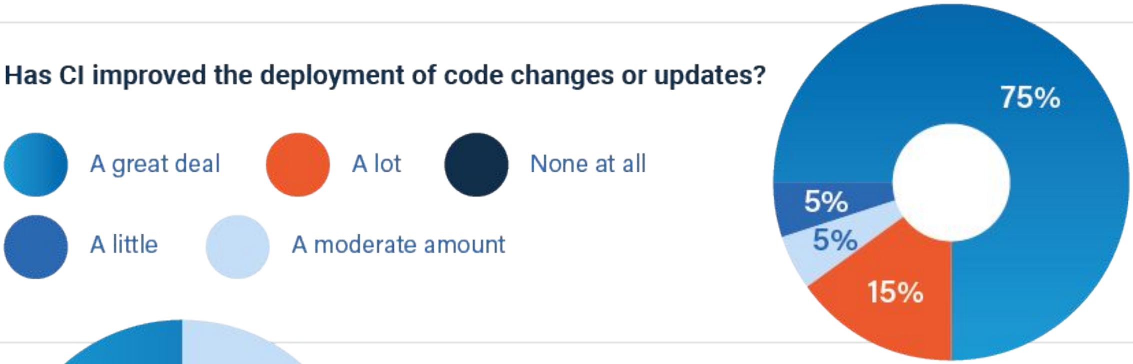
Adoption of CI best practices is on the rise

The research found that 88% of enterprise software companies have adopted CI practices, compared to 70% in 2015. As a result of implementing CI, 35% of respondents say they deploy code changes hourly. 75% of respondents say that CI has improved the deployment of code changes a great deal.

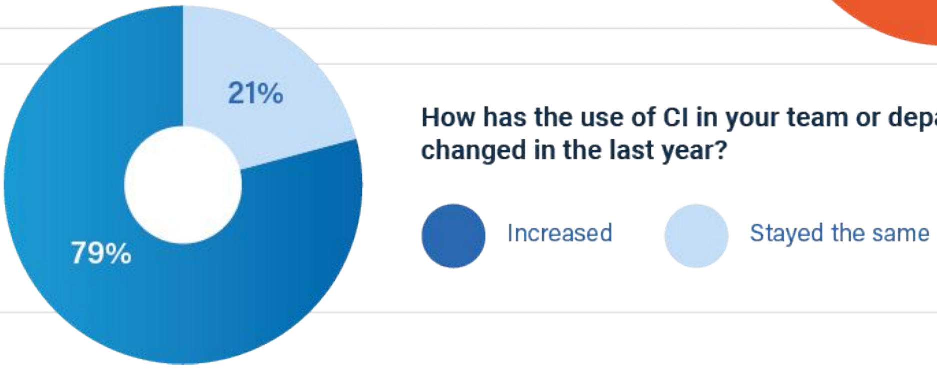
How often does your team deploy code changes/updates?



Has CI improved the deployment of code changes or updates?



How has the use of CI in your team or department changed in the last year?



SOURCE: Testing trends for 2018, Dimensional Research

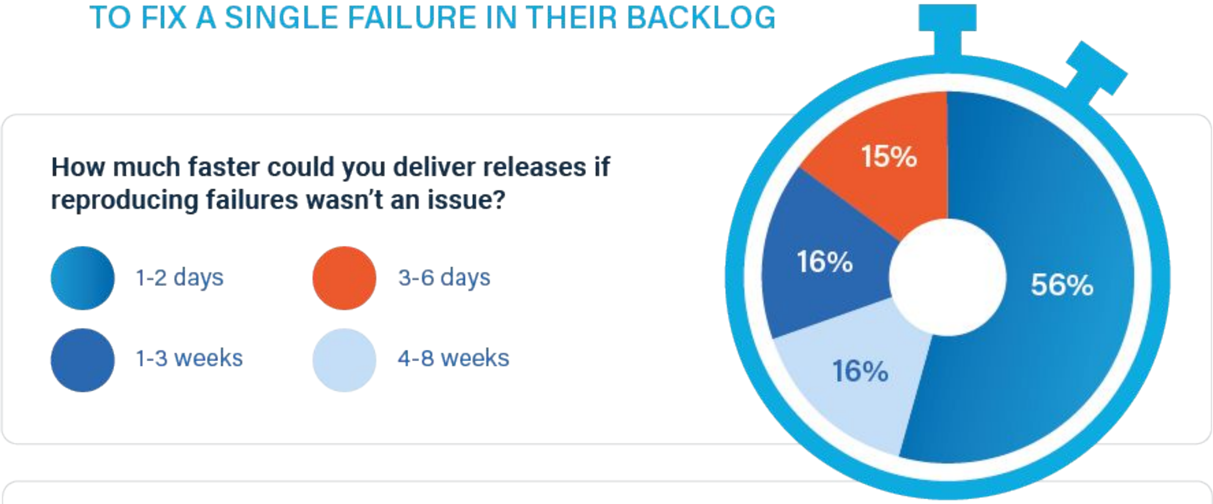
Key Finding 2

Reproducing software failures is impeding delivery speed

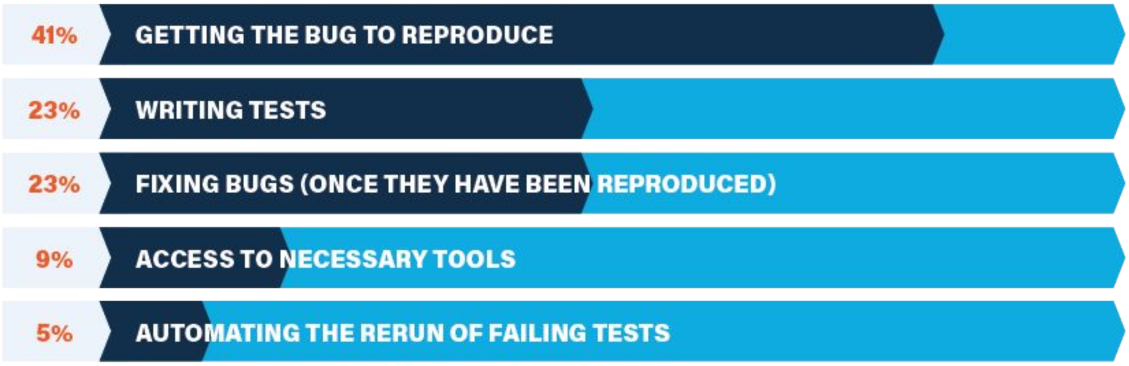
41% of respondents say getting the bug to reproduce is the biggest barrier to finding and fixing bugs faster; and 56% say they could release software 1–2 days faster if reproducing the failures wasn't an issue, while 16% say they could deliver at least 4 weeks faster.

83% of survey respondents say they cannot keep their test suites green and that the average amount of time needed to find and fix one failure in their backlog is 13 hours.

> SOFTWARE ENGINEERS SPEND AN AVERAGE OF
> **13 HOURS**
> TO FIX A SINGLE FAILURE IN THEIR BACKLOG



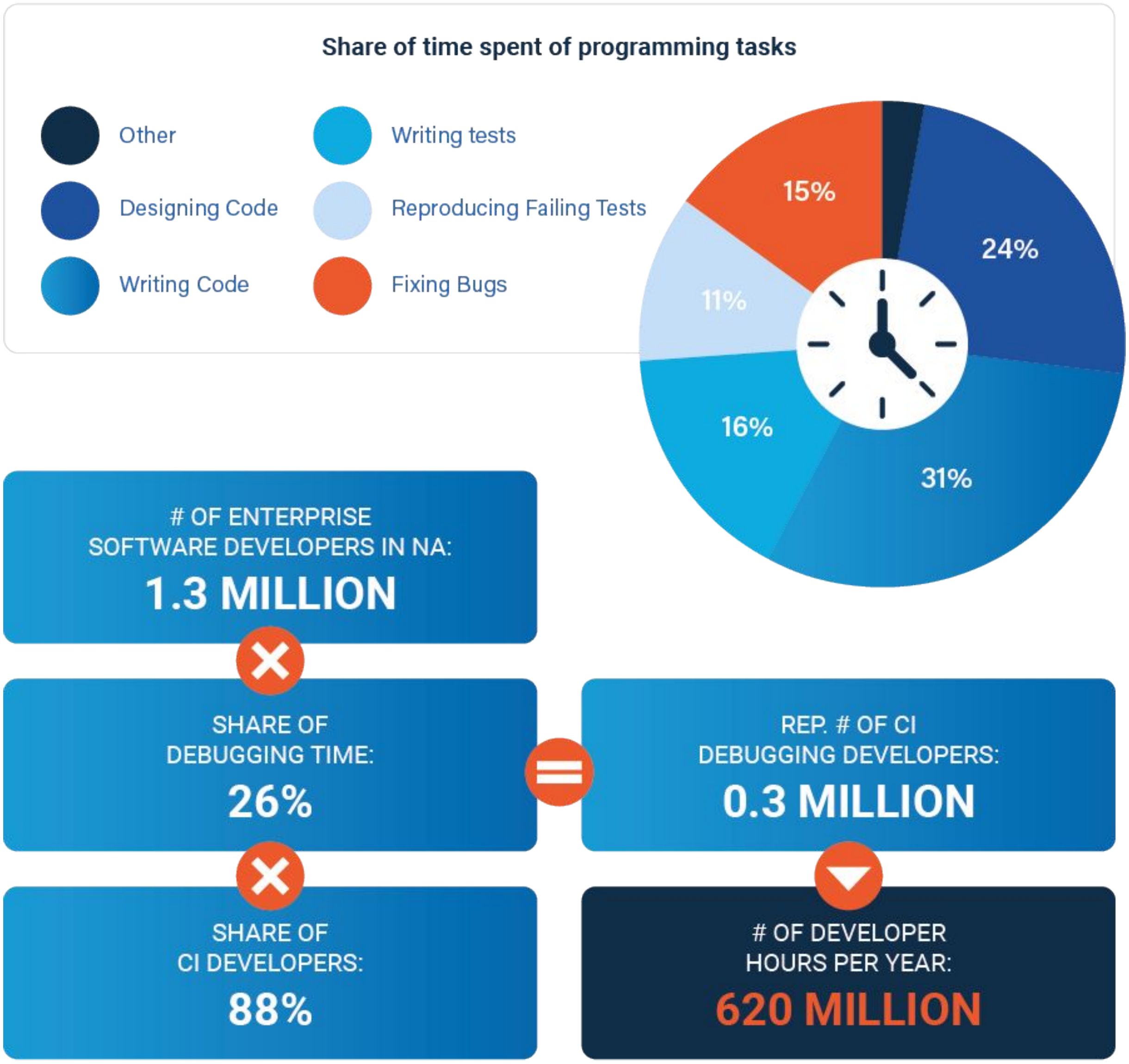
What is the biggest barrier to finding and fixing bugs in your backlog faster?



Key Finding 3

Failing tests cost the enterprise software market $61 billion annually

According to the research, 26% of developer time is spent reproducing and fixing failing tests – which equals 620 million developer hours a year. The total value of salary spent on those hours adds up to $61 billion annually and equates to $1.2 trillion in enterprise value lost for shareholders per year.



OF ENTERPRISE SOFTWARE DEVELOPERS IN NA: 1.3 MILLION

×

SHARE OF DEBUGGING TIME: 26%

×

SHARE OF CI DEVELOPERS: 88%

=

REP. # OF CI DEBUGGING DEVELOPERS: 0.3 MILLION

OF DEVELOPER HOURS PER YEAR: 620 MILLION

SOURCE: How many Developers are there in America, and where do they Live?, DQYDJ.com (2019)
SOURCE: Software Development Productivity: Most Important Metrics to Track. Stackify.com (2019)

Conclusion

In a world where every company is a software company, the ability for software engineering teams to deliver high-quality software at speed has become a primary competitive advantage. But software failures are inevitable. As teams add new features, new processes, new microservices, and new threading to their code, the risk of unpredictable failures grows exponentially.

This research conducted by a Cambridge Judge Business School MBA project sought to understand the financial implications of those software failures in development and test. The findings are clear: **software defects are expensive and reproducing software failures is impeding delivery speed.**

True CI requires a zero-tolerance approach to software failures. Tests *must* pass reliably and any failures represent new regressions. Intermittent failures that only show up once every 300 runs, or under extreme conditions, only make this more challenging.

Time travel debugging fixes that problem entirely. This capability can be integrated into your test suite to automatically generate recordings of any failed test, so no time is spent trying to reproduce test failures. Developers simply debug the recording (akin to video footage) to rapidly locate the root causes of a new regression, legacy bugs, and flaky tests.

Learn more about time travel debugging

Market Size

For the purposes of this analysis, the study focused on software engineers across various industries and job levels.

As of 2018, the enterprise market in North America reached a value of $180.5 billion, an 8.8% increase over the previous year. The United States accounted for 90%, or $162 billion, of the North American enterprise software market. With CAGR growth of 15.3%, the software enterprise market is the fastest growing segment in the IT industry and projected to reach $368.6bn in 2023.

SOURCE: Software in North America, MarketLine (2019)

Methodology

Respondent job titles and functions were spread across software development, architecture, DevOps, QA and automation engineering. Respondents were surveyed through 1-on-1 interviews and through an emailed survey and were picked based on industry and geographical fit. The final sample size was 73 and the respondents were not offered any compensation for completing the survey.

Participants were asked to answer questions about the perceived costs of software defects caused by failing tests in CI pipelines. Specifically, they were asked to assess the quantitative and qualitative impacts of failures in the software development pipeline, and to determine the costs that could be saved if they were able to resolve those issues faster.

Undo is the time travel debugging company for Linux. We equip developers with the technology to understand complex code and fix bugs faster.

Developers spend far too much time figuring out what code actually does – either to find and fix bugs or to understand other people's code. With time travel debugging, developers can see exactly what the software did, and create better software faster.

By integrating Undo's LiveRecorder into their CI workflow, C/C++, Go, and Java software teams can easily and quickly diagnose the root causes of new regressions, legacy bugs, and flaky tests – before they hit customers.

Thousands of developers across leading technology firms including SAP, Juniper Networks, and Siemens use LiveRecorder to improve developer productivity, developer velocity, and software quality.

https://undo.io
info@undo.io