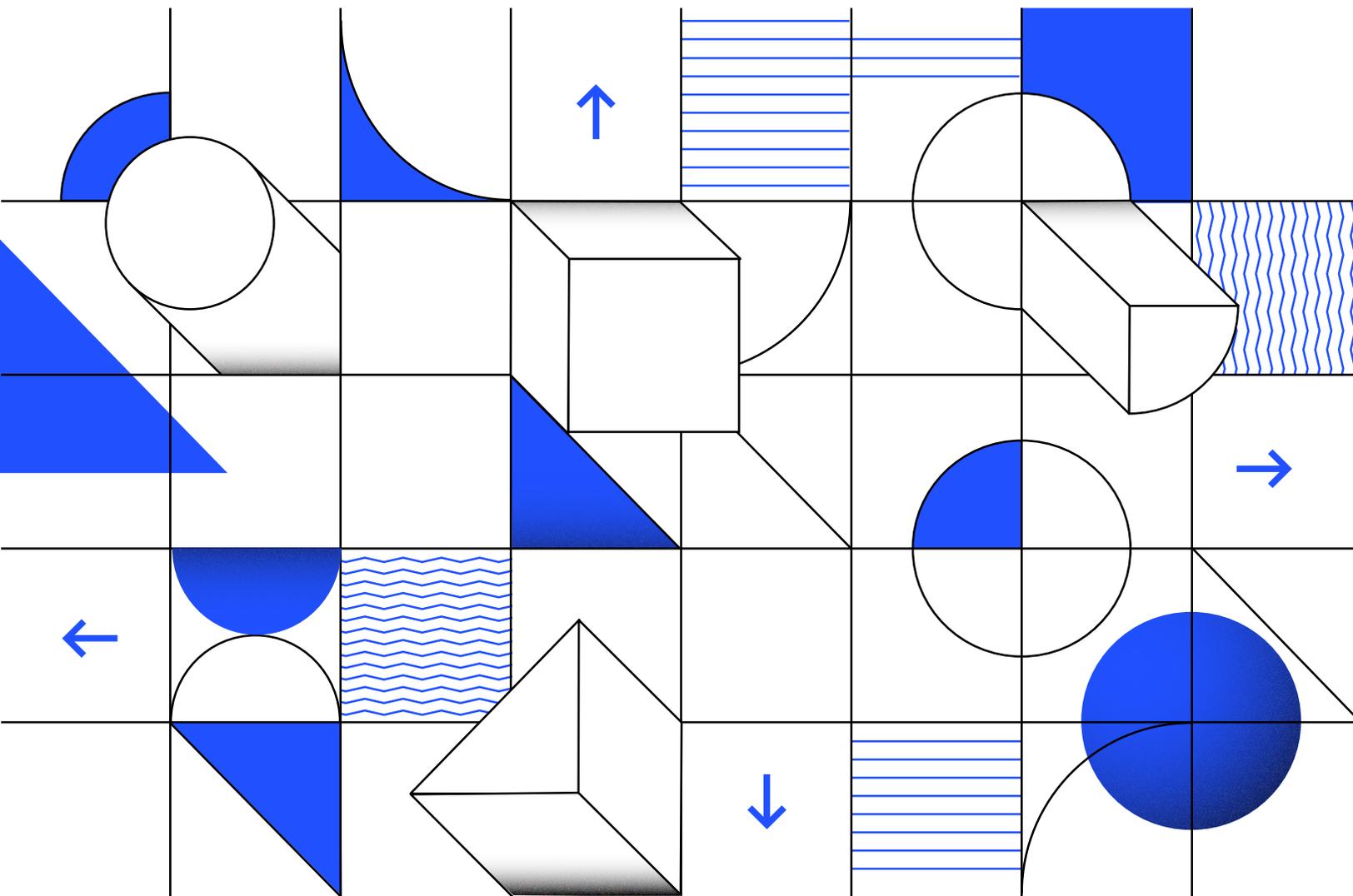


**DX**

# Developer Experience Management: The Key to Building High Performing Engineering Organizations

The strategies for building high-performing engineering organizations have continually evolved. Today, the world's most innovative companies are maximizing developer productivity and engagement by focusing on the developer experience.



## Introduction

As costs and competition for engineers continue to rise, companies are more focused than ever on removing points of friction in order to retain their talent and maximize the effectiveness of their developers.

While companies have always invested in tools and capabilities to support their developers, what's different today is the manner in which companies are listening to their developers—treating them as they would customers—to systematically identify and address the biggest areas of friction in the software development lifecycle.

Industry leaders<sup>1</sup> refer to this practice as “Developer Experience Management.” This whitepaper outlines the importance of developer experience and presents a new framework for measuring it.

**Developer Experience (DX)** refers to the day-to-day experience of developers, affected by the tools and processes that they encounter in their daily work.

**Developer Experience Management** is the process of removing friction, empowering developers to be as productive and engaged as they can be.

---

1. Companies including Stripe, Spotify, and Chime have recently formed dedicated internal Developer Experience teams.

# How Developer Experience Impacts the Bottom Line

Developer experience isn't just a feel-good investment. Research shows that better developer experience helps engineering organizations achieve better efficiency, talent retention, and business performance.

## Increased efficiency

Interruptions and slow feedback loops are common points of friction for developers. The 2019 State of DevOps report found that developers typically only spend 30-40% time on developing features, with significant time going toward administrative tasks and delays. Small improvements to the developer experience can have significant impact, e.g., saving two hours per week per developer across an organization of 200 engineers results in annual savings of \$2,880,000 (the equivalent of 10 full-time developers).<sup>2</sup>

## Reduced attrition

Inefficient tools and processes frustrate developers and result in them leaving, especially when problems are not improved<sup>3</sup>. Studies show developer experience as the number one reason for developers voluntarily leaving their jobs, ranking above salary and benefits<sup>4</sup>. With competition for top tech talent soaring, improving developer experience is key to reducing turnover and curtailing the significant costs of hiring and lost capacity.

# 25%

Average developer time lost per week due to inefficiencies in the developer experience.

# 80k

Total cost in dollars of replacing a developer who voluntarily leaves<sup>5</sup>.

---

2. Calculation assumes 48 work weeks per year and an average hourly rate of \$150 per hour

3. Greiler, "An Actionable Framework for Understanding and Improving Developer Experience"

4. Westlund, "Retaining Talent: Assessing Job Satisfaction Facet Most Significantly Related to Software Developer Turnover Intentions"

5. Study based on composite organization of 200 developers with average salary of \$150,000

## Improved business performance

Improving developer experience is about creating the optimal environment for developers to innovate and deliver value for their teams. Companies with better developer experience not only outperform their competition in terms of productivity and their ability to innovate faster, but also business performance. A 2020 McKinsey study<sup>6</sup> found that companies that perform well at creating optimal working environments for developers achieve revenue growth that is four to five times faster than their competition. High-performing organizations also scored higher on innovation, customer satisfaction, and brand perception. The 2021 State of DevOps report similarly found that high-performers were twice as likely to exceed organizational performance goals as low performers, outperforming competitors in areas such as profitability, market share, and customer satisfaction<sup>7</sup>.

---

6. Srivastava, "Developer Velocity: How Software Excellence Fuels Business Performance."

7. Smith, D., et al, "State of DevOps 2021"

# Measuring Developer Experience: Where Traditional Metrics Fall Short

One of the complex challenges organizations face when trying to improve developer experience is deciding what to measure. Traditional engineering metrics help surface development activity but are unable to identify root causes of slowdowns and frustration. To successfully measure developer experience, a new approach is needed.

## Traditional approaches

**Measuring output.** For decades, leaders have attempted to measure developer productivity through output metrics such as lines of code, velocity points, or number of pull requests. These metrics, do not account for the reality of engineering work, ignoring factors such as quality or difficulty of work. As a result, output metrics do not provide meaningful signals and often incentivize detrimental practices (e.g., writing more lines of code results in bloated code).

**Measuring process.** More recently, leaders have gravitated toward process metrics such as lead time and deployment frequency. While useful for benchmarking software delivery processes, these metrics do not elucidate the specific bottlenecks and painpoints that developers face. As a result, leaders and teams today lack visibility into which areas of their tools and processes could be most improved.

“

The common anti-pattern is to look at lines of code, feature output or to put too much focus on trying to spot the underperforming developers. It is better to turn the conversation around to focus on how the organization is providing an effective engineering environment.<sup>8</sup>

Tim Cochran

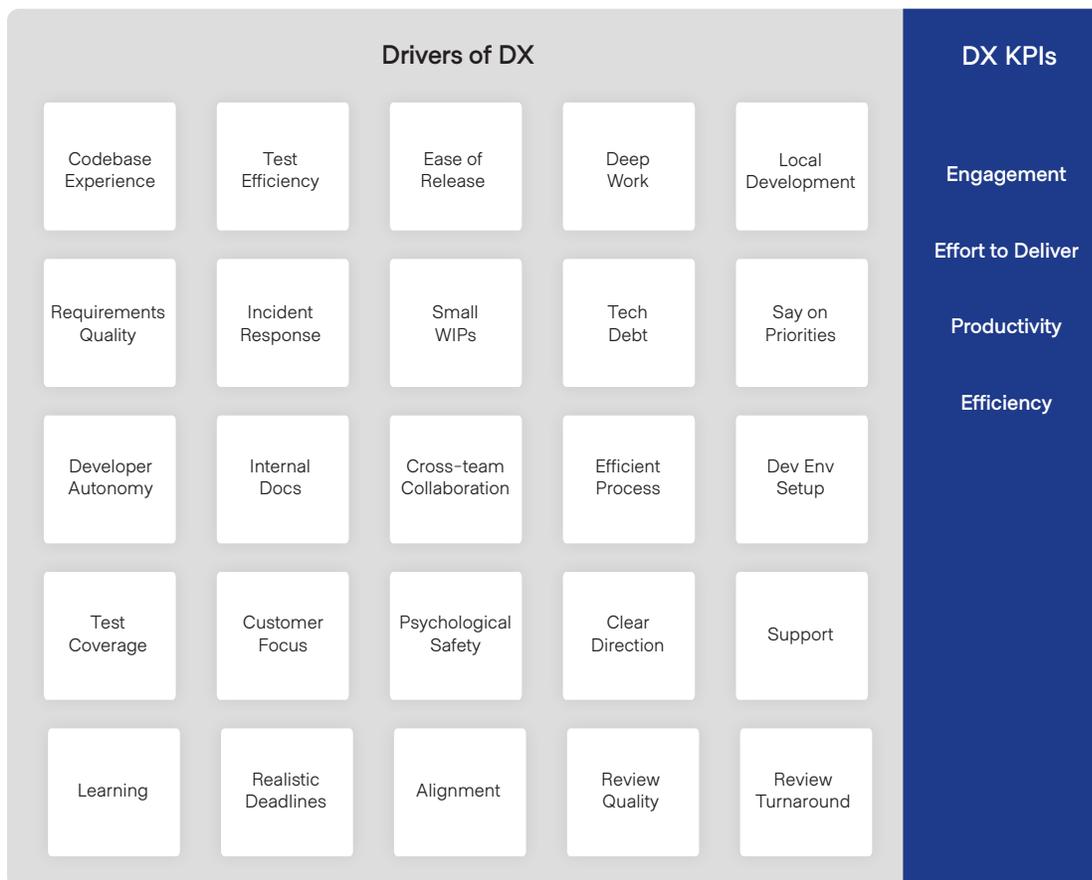
Technical Director, Thoughtworks

---

8. Cochran, "Maximizing Developer Effectiveness"

## DX25: A Scientific Framework for Measuring Developer Experience

DX25 is a new framework to help organizations measure the four core KPIs and 25 top drivers of developer experience. Based on peer-reviewed research<sup>9</sup>, DX25 provides a holistic picture of friction encountered by developers across teams, tools, and processes. Unlike traditional metrics that are generated from tools like GitHub and Jira, DX25 utilizes perceptual and behavioral measures to provide an in-the-trenches understanding of the daily experiences of developers. To learn more about DX25, visit [getdx.com](https://getdx.com).



9. Greiler et al., "An Actionable Framework for Understanding and Improving Developer Experience"

## About DX

DX is the world's first developer experience platform, helping organizations measure and improve top drivers of developer productivity and engagement. DX is designed by leading software engineering researchers, providing science-backed metrics, workflows, and education that empower teams to improve. To learn more, visit [getdx.com](https://getdx.com).