OCT 23 INDUSTRY INSIGHTS

CloudBees.

Platform Engineering in 2023:

Rapid Adoption and Impact

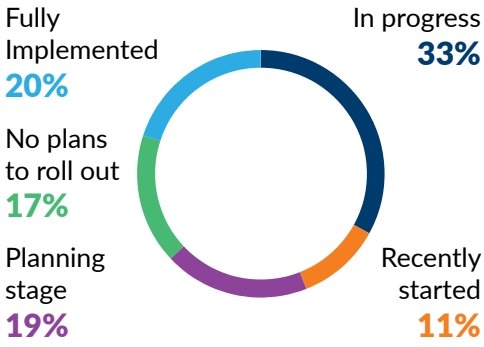
Produced By Techstrong Research

Platform engineering has quickly gained traction within IT organizations. It's fast becoming an established practice for DevOps and software development teams, as shown by our survey and other research organizations' findings.

Platform engineering emerged to address the "forgotten developer," lost with the heightened emphasis on agile, cross-functional teams and DevOps pipelines. With the economic downturn over the past 12 to 18 months, software leaders were forced to do more with less. This shifted focus to making developers more productive, but a clear path to achieve greater productivity was lacking. Some industry pundits confidently declared that DevOps was dead and pointed to platform engineering as its replacement. That couldn't be farther from the truth. In reality, platform engineering has gained its own focus and set of engineering disciplines that are complementary to DevOps.

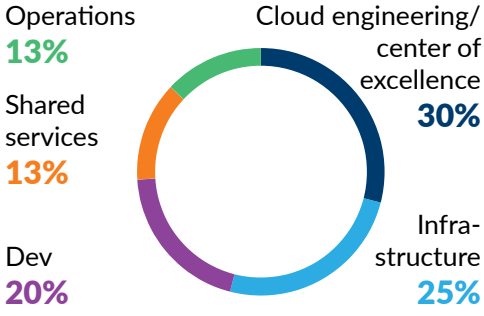
In mid-2023, CloudBees polled 221 practitioners within the software communities of DevOps, platform engineering, cloud native, and cybersecurity to take their pulse on the adoption and impact of platform engineering in their organizations. The data shows very significant levels of adoption (83%), with 20% fully adopted, 44% in progress or recently started, and 19% in the planning stages (Figure 1). Only 17% of respondents had no plans for adoption. One way platform engineering shows its adaptability is where the function lives in an organization. Platform engineering has the strongest presence in cloud engineering (30%) and infrastructure (25%) departments, followed by development (20%), shared services (13%), and operations (13%) (Figure 2).

Figure 1: Where is your organization in adopting platform engineering?



Platform engineering has made major inroads with a substantial percentage (83%) indicating they are in the adoption mode. Platform engineering is very much here to stay.

Figure 2: Where does the platform engineering team sit in your organizational structure?



Platform engineering's home in organizations varies greatly. Cloud engineering and infrastructure groups are more common, though development, ops, and shared services are also viable options. Platform engineering is likely placed where it makes the most sense for each organization.

Our research shows the initial drivers to adopt platform engineering were evenly distributed across a desire to improve developer productivity, implement CI/CD pipelines, standardize tools/processes, enhance security, and manage infrastructure-as-code (Figure 3). The ability of platform engineering to adapt to the specific needs of each organization is essential to its acceptance. Like DevOps, there is no predefined path for implementing platform engineering. Rather, the organization will implement and use platform engineering in ways that address its highest priorities (Figure 4).

Because of the complementary nature of platform engineering and DevOps, the emergence of platform engineering is timely. Platform engineering takes a holistic view of developers and the environments they work in and establishes practices around internal developer platforms (IDPs). IDPs are, in turn, defined around improving developer experiences (DevEx). The rise in platform engineering directly addressed hits to software dev cycles due to the complexities of IDEs, plugins, toolchains, repositories, environment creation, and incompatibilities.

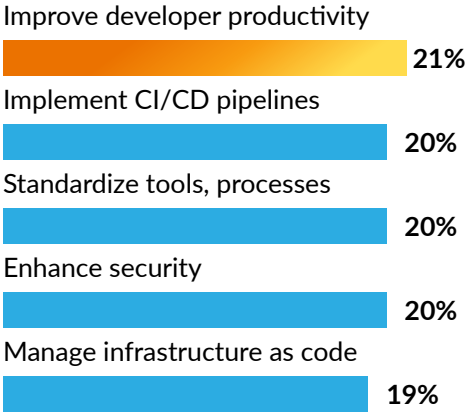
In fact, virtually all of the platform engineering objectives rated most important relate to DevEx and improving productivity for developers. The three highest-ranked objectives were self-service for developers (29%), easy adoption (25%), and meeting developer needs (20%) (Figure 5).

Finally, the measures of success for platform engineering teams run the gamut from developer productivity (the highest-ranked measure, at 23%) to internal DevOps KPI attainment, cost control, reigning in tool sprawl, simplifying the environment, security, and stability improvements (Figure 6).

2024 and Beyond

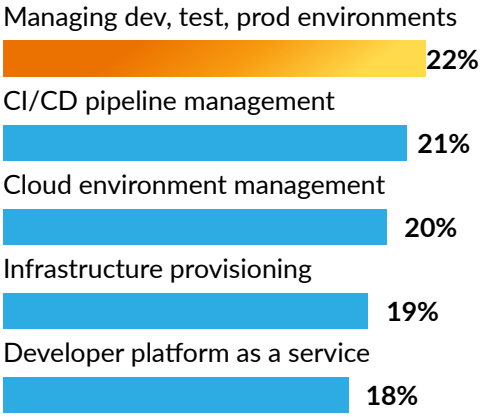
What's in store for platform engineering in the next 12 to 18 months? First, it's important that platform teams equip themselves for ongoing, continuous change. Companies are acquired, new platforms are acquired (often without shedding old ones), and new applications and deployment patterns will continue to emerge.

Figure 3: What drove your organization to adopt platform engineering?



There is no singular reason organizations adopt platform engineering. It is appealing as a way to solve multiple pain points across software development, workflow pipelines, infrastructure, and security.

Figure 4: What are the main uses of platform engineering in your organization?



Organizations invest in platform engineering to address multiple needs. Aligned with the developer productivity focus, CI/CD pipeline and dev/test/prod environment management come out on top. There is no one-size-fits-all use case for every organization.

The most recent disruptive technology to arrive is generative artificial intelligence (AI), following closely on the heels of AI and particularly machine learning (ML). AI can and will be applied to improving platform engineering. However, like every other aspect of creating, operating, and securing software and systems, AI/ML and generative AI bring with them challenges. We see three significant ones:

The first is managing multiple large sets of data and models, the lifeblood of ML algorithms and generative AI large language models (LLMs). Like specialized expert systems, domain-specific LLMs trained on internal enterprise data will prove particularly important in adopting generative AI, provided data privacy and security are maintained.

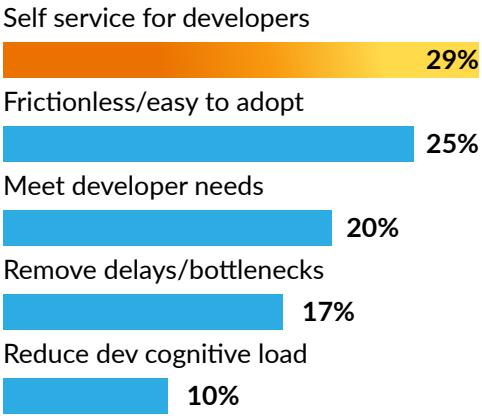
Second, platform engineering must adapt to new AI workflows and pipelines for data, prompts, and the AI engineers who design, train, and maintain models, vector databases, and large datasets as they grow and evolve. These AI pipelines must support particulars of their workflow patterns and coincide and integrate with interdependent software development pipelines and release processes.

Lastly, AI/ML and generative AI often have operating characteristics separate and apart from the cloud and application environments we understand and operate today. AI brings new hardware operating environments, including AI accelerators, GPUs, VPUs, and highly scalable CPUs, and challenging performance and optimization learning curves. Platform engineering will play a crucial role as AI, particularly generative AI, is adopted and mainstreamed in enterprises.

221 practitioners were surveyed in August 2023. Approximately 54% were from North America, 19% from EMEA, 19% from APAC, and 8% from LATAM. 36% worked in organizations that had 100 developers or less; 23% worked in organizations that had 101 - 1,000 developers; 22% work in organizations that had 1,001 - 10,000; and 19% had 10,001 - 25,000+ developers.

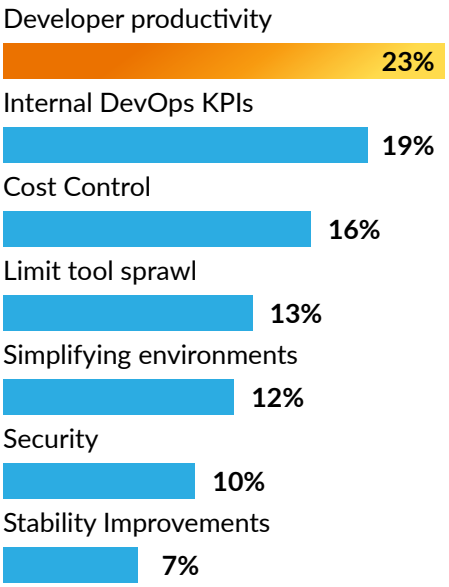
1023v00

Figure 5: What platform engineering objectives are most important to your organization?



As DevOps grew in popularity, platform engineering's rise to prominence was often attributed to the "We forgot about the developers!" phenomenon. While there are many positives, developer productivity strongly influences the creation of self-service tools and infrastructure, eliminating friction and bottlenecks and creating "developer-friendly" environments.

Figure 6: How do you measure platform engineering success?



Developer productivity is a commonly accepted driver of platform engineering adoption. But there's more to the story, with internal DevOps KPIs and cost controls following closely as important determinants of success.

Techstrong | Research