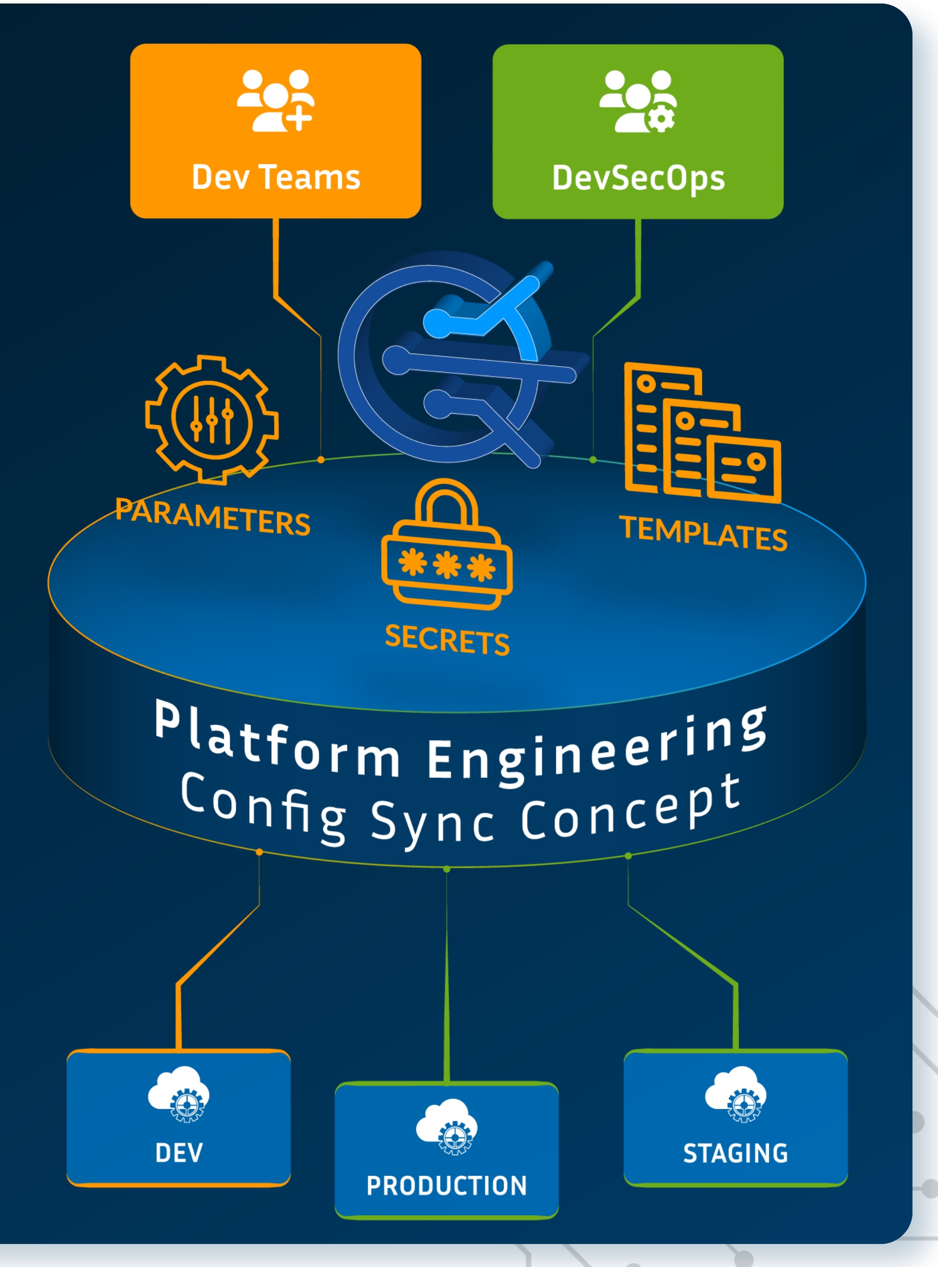
CLOUDTRUTH

The Future of Platform Engineering:

Trends and Predictions



One “constant” in our world is that technology is constantly evolving. New tools, systems, and procedures create opportunities to “do better faster,” but these advances come with a cost. The need for speed and the increasing complexity of modern software architectures has made platform engineering a hot topic in the Ops and cloud infrastructure space.

The past decade has brought a seismic shift in how organizations approach their application deployments. The fast rise of containerization, microservices, and application modernization leads to configuration sprawl, which impacts an organization’s ability to build, deploy, and manage its applications at scale.

Platform engineering improves the developer experience, bolsters productivity, and empowers companies to stay competitive in an increasingly digital world. The future of platform engineering is evolving quickly, and organizations need to understand how to utilize this emerging concept and stay ahead of the curve.

Let’s take a high-level look at platform engineering and then focus on some proven best practices. Finally, we’ll examine future platform engineering trends and predictions we expect to see in the coming years.

What Makes Platform Engineering a Game-Changer?

First, let's start with a basic definition. **Platform engineering refers to the design, development, and maintenance of a self-service internal software platform that enables the efficient creation, deployment, and management of applications at scale.** Created to account for increasingly sophisticated IT environments, platform engineering provides a set of tools, frameworks, and services that can be used by developers to build, test, deploy, and manage applications with confidence, speed, and control. Together, this combination of solutions is often called an Internal Developer Platform (IDP).

Essentially, platform engineering gives developers a standardized way of managing application data, workflows, and infrastructures, all while making it easier for organizations to scale their apps and remain agile as business needs evolve.

Platform engineering is undoubtedly a hot topic in the developer space today, and for good reason. It's already been proven to enhance the developer experience and make it easier for them to overcome configuration complexity challenges.

THE BENEFITS SPEAK FOR THEMSELVES:

Reduces the time and cost to ship new features

Facilitates collaboration between App Devs and DevOps

Minimizes the risk of misconfigurations

Streamlines the complete development process

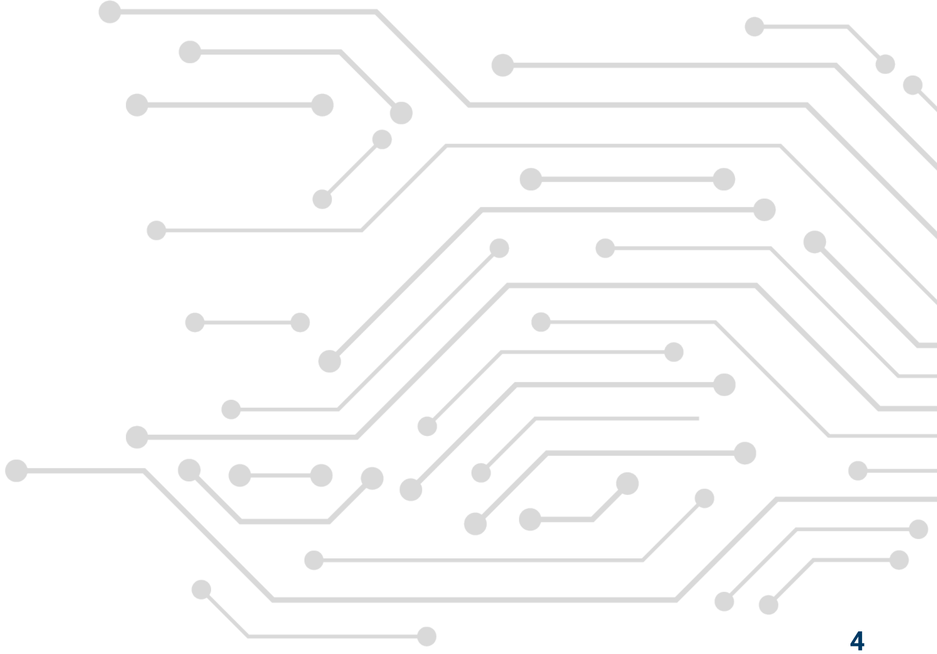
Enables companies to respond quickly to changing business needs

59% of software engineers consider platform engineering to be critical for their organization’s success.

65% of organizations use platform engineering to increase productivity and reduce time to market.

Organizations that implemented a platform approach to application development and delivery saw a 50% reduction in the time required to deliver new applications.

Think of platform engineering as building blocks, best practice patterns, and blueprints to align all teams on the same methodology to build and ship software. When Dev and Ops are aligned, release velocity increases dramatically.



Building an A-Team: Secrets to Platform Engineering Success

Getting platform engineering right largely hinges on your ability to assemble a team that effectively automates infrastructure management and enables developers with self-serve tools. Luckily, some tips and tricks can make it easier to build the best platform team possible and, in turn, get the most out of the Internal Developer Platform.

▶ DEFINE YOUR OBJECTIVES

Setting a clear mission for your team and ensuring everyone knows their role in your platform engineering initiatives will deliver the best results.

▶ TACKLE THE COMMON CHALLENGES FIRST

Your platform team should identify the common pain points across all the teams and provide solutions to remove the friction.

▶ DESIGN FOR SCALABILITY FROM THE GET-GO

Implementing a forward-looking platform engineering strategy will help avoid costly refactoring and support a growing organization with a solid foundation.

▶ TREAT YOUR PLATFORM LIKE A PRODUCT

Developer portals serve an internal customer: application developers. A product manager should own the roadmap, solicit feedback, triage the backlog, and identify goals and KPIs.

▶ STANDARDIZE, STANDARDIZE, STANDARDIZE

One of your platform team’s goals is to streamline developer workflows. That’s not easy when they take different approaches across siloed teams. Standardizing engineering processes and tools increases buy-in from your platform team and enables them to act as a collective unit.

How Will Platform Engineering Shape the Future of DevOps?

Platform engineering is relatively new, and its growth has accelerated significantly. With no signs of slowing down, it's fair to assume that platform engineering will coexist with DevOps teams for some time.

Industry experts predict these four trends for platform engineering in 2023 and beyond.

BLUEPRINTS FOR SUCCESS WILL START TO CIRCULATE

Platform engineering, a novel concept, initially left early adopters to devise their own best practices for success. As more teams adopt developer portals, established blueprints and patterns will emerge, benefiting others and accelerating adoption. This virtuous cycle will attract even more teams to embrace platform engineering.

There are 11,000+ active members in the Platform Engineering Slack community, up from 1,000 in 2022.

SECRETS AND CONFIG MANAGEMENT SET FOR MAJOR ADVANCEMENTS

The intersection of platform engineering, secrets, and config management is a critical area of focus for organizations seeking to maintain secure and efficient development environments. Platform engineering provides developers with consistent, automated infrastructure, streamlining the development process while minimizing operational overhead.

Expect new config and secrets management techniques to emerge as platform engineering matures.

Gartner predicts that 80% of software engineering organizations will establish platform teams by 2026 as internal providers of reusable services, components, and tools for application delivery.

DEVELOPER-FIRST PLATFORMS MINIMIZE K8S COMPLEXITIES

When Kubernetes (K8s) came onto the scene, organizations really had no choice other than to adopt an ops-first approach to deliver software. But in recent years, we've seen companies adopt a "shift-left" mindset and empower their developers to address Kubernetes challenges quickly and efficiently.

In 2023, organizations that build and embrace developer-first platforms will have an easier time overcoming Kubernetes complexities and improving productivity across their teams.

This doesn't mean developers need to be tasked with managing the full software life cycle, but **they should be empowered with self-service tools and resources in order to speed up your time to market and enable faster feedback loops.**

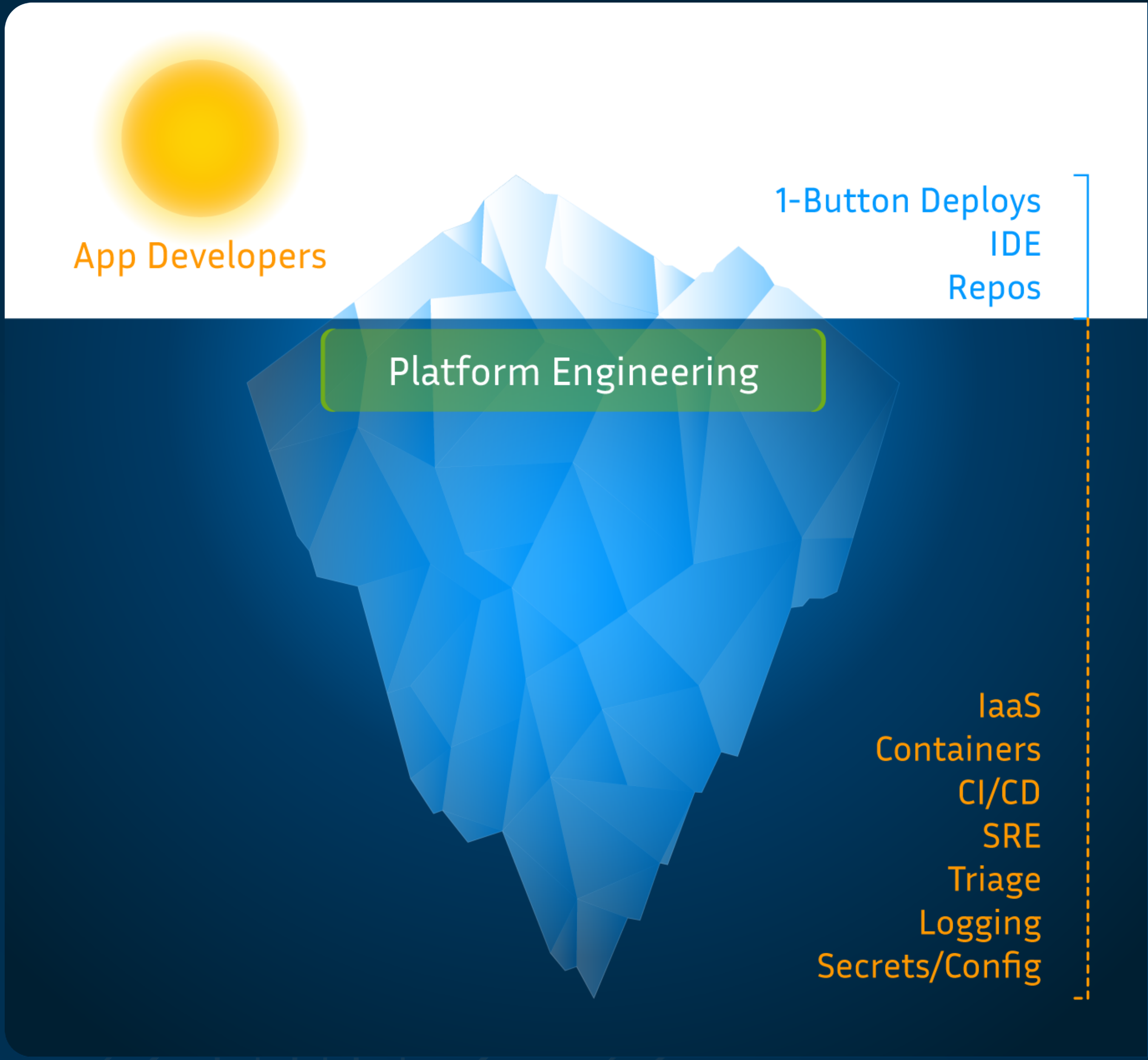
A study by McKinsey found that companies that prioritize developer experience and invest in developer platforms are more likely to achieve their digital transformation goals and see increased revenue growth.

DEVOPS ROLES WILL EVOLVE

Organizations are constantly looking to empower their developers to do more and stress less. This shift-left approach has developers handling a wide range of responsibilities, from observability and security to scalability and operations. This begs the question — if developers are now being tasked with building platforms for themselves, is a DevOps team really necessary?

The answer is yes. In order for developers to do their best work, they need a DevOps team that's able to provide them with the right tools, systems, and resources. As developer platforms become the new norm, the role of DevOps will need to evolve, not fade away. Ideally, developers will be able to seamlessly shift left without being overburdened with responsibilities that are outside their wheelhouse.

84% of organizations believe that DevOps enables them to deliver software faster and with higher quality, and 82% believe that DevOps improves their ability to innovate.



Get Started With Free Config Observability

One area DevOps, platform engineers, and application developers can agree upon is a need for a self-service configuration management platform to remove the toil associated with managing secrets, parameters, and ENV variables.

Streamlining deployments, increasing release velocity, and improving operational efficiency across the board is much easier when your teams have complete visibility of all their config. In short, configuration observability enables DevOps and platform engineers to monitor and manage system configurations more effectively and identify issues quickly while improving your platform's reliability, availability, and security.

This is where CloudTruth comes in.

Our free-forever config observability platform provides a centralized location to manage and view all secrets and parameters in one place. Get versioned history, config rollbacks, and change tracking with RBAC to protect sensitive information.

Our free config observability features also include:

- Five users and three integrations
- 7-day audit and history retention
- SSO with Google, Microsoft and GitHub
- Community support (Discord)

Platform engineering is shaping the future of configuration management and application deployment. Get ahead of the curve and let CloudTruth empower your DevOps personnel and platform engineers to do their best work. Sign up for your free account today!

GET STARTED WITH CLOUDTRUTH