



AHEAD

Growth & Adoption for Platform Engineering at Scale

More than a Technology Challenge

As businesses increasingly rely on innovation to serve and retain their customers, they face complex challenges in managing the exponential growth of data and technology while ensuring seamless development progress and user adoption. The problem is evident—data overload, technological complexity, scalability issues, and the struggle to drive user adoption are all too common.

With these looming challenges, a transformative solution is surfacing via **platform engineering**. According to [Gartner's 2022 Hype Cycle](#), platform engineering emerged as the leading technological trend for 2023—and for good reason. Platform engineering has become invaluable for businesses as they develop and maintain their technological platforms to support their core operations.

However, scaling platform engineering is not only a technical obstacle; it is a meticulous process that requires attention to people and procedures, as well as technology. In this whitepaper, we'll dive deeper into the interplay between these three factors, stressing how growth and adoption strategies are just as important as technical prowess regarding platform engineering at scale.

The Evolution of Technology in Platform Engineering

Recently, we have seen huge developments in platform engineering as the environment has changed steadily since the advent of microservices, serverless computing, and container orchestration. Organizations have constructed and operated complex platforms with increased efficiency and flexibility due to the widespread adoption of cloud-native technology and DevOps processes, which have sped up this transformation.

Take, as an example, the shift from monolithic to microservice-based software. Microservice design promotes adaptability and robustness by letting teams simultaneously work on separate parts of an application. This change illustrates how technology continues to evolve to meet the needs of large-scale platform engineering.

Now, let's examine the path to expansion and acceptance by analyzing the role of people, processes, and technology in enterprise platform development.

People: The Human Element

The people behind a technology project are the ones who will ultimately determine its fate. The human element is crucial in large-scale platform engineering because it involves a wide variety of roles and responsibilities, from programmers and operators to architects and managers.

Enablement for Platform Engineering

To accelerate widespread platform engineering adoption, businesses must equip their employees adequately. In doing so, one of the very first steps should include creating a plan for change management and execution. Introducing technological changes can be difficult within large enterprises, and a thorough strategy for managing inevitable changes throughout the transition is crucial. The proper implementation should look something like this:

Organizational Change Management & Communications

Change Management Design and Execution

Drive the Expansion and Adoption of Platform Engineering | Continuously Accelerate the Speed at which IT does Business

Aligned Management, Standardization, Reporting

- Standard Reporting
- Continual Service Improvement
- Budget and Finance
- Project Integration
- Accountability and Ownership

Technology Alignment and Integration

- Architecture Support
- Technology Integration
- Business & Technology Alignment
- Self-Service
- Reduction of Technical Debt

Change Management and Strategic Communications

- Standardized Communications
- Alignment and Awareness
- Consistency and Cadence
- Continuous Integration
- Metrics

Org Change Management >

Align stakeholders for standardization and expand capacity to rapidly adapt to change while maintaining service delivery quality. Impact index and scorecard to measure adoption.

Stakeholder Engagement	Training & Upskilling	Strategic Communications
Organizational Alignment	Measurable Success / KPIs	Measurable Adoption & Proficiency

In addition to change management, the adoption of platform engineering necessitates the following considerations:



Training & Professional Growth

Updating and expanding team skillsets should be an ongoing process. Teams must understand and use recent technological developments, tools, and best practices to avoid roadblocks. Building a Community of Excellence (CoE) or Dojo which promotes certifications, seminars, and other forms of training will play an important role in this context.



Innovation Enablement

Motivate your staff to come up with creative solutions. Platform engineering problems can be addressed in new ways if a culture of creativity and innovation is fostered across the organization.



Analyzing & Preparing

Start by taking stock of things and determining what needs to be altered. Define the transition's goals, scope, and timing.



Working Together Across Departments

Inspire inter-departmental collaboration and communication. Creating an atmosphere of cooperation is essential, as silos can impede growth. This can be facilitated by using collaboration tools and platforms like Slack and Microsoft Teams, but must also be emphasized culturally throughout the organization.



Leadership & Mentoring

Find potential leaders and help them grow within the company. Programs that pair more seasoned engineers with younger engineers facilitate learning and information transfer.



Instruction & Help

Employees should be given the time and resources to learn about and adjust to the new platform engineering processes and technologies. Make widespread use of the reference materials and documentation that you create.



Instruction & Help

Employees should be given the time and resources to learn about and adjust to the new platform engineering processes and technologies. Make widespread use of the reference materials and documentation that you create.



Progressive Deployment

Modifications should be released in stages rather than all at once. This lessens the potential for disturbance and makes it possible to adjust according to actual usage.



Acknowledgment & Reward

Recognize and reward those who accept and aid in achieving the change endeavor. Morale and drive can each benefit from occasional praise.



Streamlined Procedures

Successful platform engineering at scale relies heavily on efficient methods. The platform's stability and scalability depend on these procedures being crafted with cooperation and efficiency.



Pilots & Testing

Launch small-scale pilot projects to put novel platform engineering methods through their paces. Collect user input, then act on it by making necessary changes.



Control & Assessment

Be vigilant while developing the transformation plan. The platform engineering skills of the company should be assessed on an ongoing basis, and any necessary changes should be made quickly and decisively.



Adaptive Procedures

Agile methods like Scrum and Kanban can help platform engineering teams function more effectively. These approaches advocate for incremental improvement, continuous evaluation, and responsiveness to market shifts.

Platform Engineering Best Practices

DevOps Practices

DevOps is centered around the idea that the development and operational groups should work together. DevOps is a set of practices and tools streamlining software distribution and handling infrastructure via technology, continuous deployment, and continuous delivery (CI/CD) pipelines.

According to Humanitec's report, "The State of Platform Engineering," many professionals who call themselves "DevOps Engineers" are currently focusing on platform development.

The Concept of Containers and Orchestration

Regarding deploying and managing applications, containerization technologies like Docker and container orchestration systems like Kubernetes have been game-changers. They make managing the platform's expansion easier by allowing consistent and scalable deployments. These solutions also introduce a moderate-to-significant degree of complexity. Establishing sound platform engineering practices is essential to scaling operational excellence.

Infrastructure as Code (IaC)

IaC allows platform developers to use code to create and manage infrastructure. This method assures that resource provisioning and configuration are consistent, repeatable, and scalable.



Technology: Building a Solid Foundation

While technology is not the only factor in determining success for large-scale platform engineering, it is still important to carefully pick the appropriate technology and techniques. Some key technological considerations include:

Technologies Born in the Cloud

Regarding platform engineering at scale, cloud-native solutions like AWS, Azure, and Google Cloud can provide the scalability and flexibility needed for enterprise-level adoption. These frameworks provide comprehensive tools and resources for creating and maintaining software programs.

Serverless Computing

Developers are free from worrying about underlying infrastructure in serverless computing. For certain kinds of employment, it's a great option because it cuts down on administrative work.

Compliance & Safety

The platform's security should be designed from the beginning. In today's threat scenario, best practices and compliance procedures must be implemented for security and maintained on an ongoing basis.

Microservices Architecture

As mentioned earlier, microservices allow groups to create and control smaller, more autonomous services. These services can be upgraded and expanded independently, increasing the platform's adaptability and robustness.

Observability & Monitoring

The capacity to detect and address problems as they arise in real time depends on robust observability and monitoring infrastructure. The performance of a platform can be thoroughly analyzed using tools like Prometheus, Grafana, and Datadog.

The Interplay of People, Process, and Technology

Simply making improvements to staff, procedures, and technology is not enough to guarantee the success of a large-scale platform engineering initiative. Rather, it is the complex and coordinated interplay between these three factors that ultimately determines what can be achieved. Below is a look at a few examples where this interaction is clear:

Case 1: Adapting New Technologies

Let's pretend a company has settled on Kubernetes as its preferred container orchestration platform. While the technology can be impressive, human and organizational factors are crucial to widespread adoption.

Technology

For Kubernetes to be fully implemented, DevOps and CI/CD pipelines must be modernized. To make the most of containers, applications that have already been developed need to be modified or re-architected.

People

Training is necessary for engineers to master Kubernetes. Architects must provide solutions that are compatible with Kubernetes. Changes in how things are deployed and monitored require operational teams to adjust.



Case 2: Enhanced Procedures

Let's say a group of engineers working on a platform will adopt Scrum or another type of Agile development. This influences both humans and technology:

Technology

Scrum processes require new or revised project management, communication, and code versioning technologies.

People

Team members need instruction on the Scrum framework and its rituals. Managers need to adopt a process that is more iterative and collaborative.

Case 3: People Development

At a company that values education, engineers are encouraged to acquire expertise in cloud-native systems. As a result, technology and procedure evolve with their workforce

Technology

To facilitate ongoing education within the workforce, cloud-based applications and resources must be made available for experiential training. It allows for the establishment of test beds and sandboxes.

People

Organizational culture now prioritizes learning and growth. Cloud-native technology adoption is boosted when engineers can use their newly acquired knowledge on actual projects.

A Holistic Strategy for Large-Scale Platform Engineering

Scalable platform engineering is a journey that spans people, procedures, and technology. Those who find success in this endeavor put resources into empowering employees, creating streamlined operations, and managing complex technological stacks.

Optimizing Resilient Technology Operations



Management of complex systems through logging, monitoring, and tracing is mandatory for delivering better software. Observability enables Platform Engineers to build resilient systems for Digital Producers



Cross-functional teams aligned with business value streams. They define, build, and release artifacts which produce a business outcome.



Core team and rotating members from delivery build, deploy, and operate production with automation, reliability, scalability, and speed.



Align stakeholders for standardization and expand capacity to rapidly adapt to change while increasing service delivery quality.

Growth and adoption in platform engineering are not conventional, but iterative processes—understanding this is crucial. The keys to success include well thought-out change management strategies, rigorous self-analysis, and a flexible approach. Today's platform engineers must be adaptable and open to new ideas and methods to keep up with the rapid pace of technological change.

Platform engineering at scale is not merely a technical obstacle, but a strategic undertaking that can revolutionize the entire enterprise if handled with the proper frame of mind. Organizations can set out on a path that will make them leaders in the rapidly changing field of technology and platform engineering by developing a constant enhancement culture and harmonizing these three elements.



To learn more, reach out to [AHEAD](#) today.

Contributing Author:

Stephen King, Chief Architect, Digital Solutions

About AHEAD

Combining cloud-native capabilities in software and data engineering with an unparalleled track record of modernizing infrastructure, we're uniquely positioned to help accelerate the promise of digital transformation.

National Hubs

ATLANTA

1117 Perimeter Center
W406
Atlanta, GA 30338

CHICAGO

401 Michigan Ave.
#3400
Chicago, IL 60611

SAN FRANCISCO

2000 Crow Canyon Place
Suite 250
San Ramon, CA 94583