LocalStack

Whitepaper

Understanding Local Cloud Development for Enterprise Software Organizations

Introduction

The value that enterprises are getting from the cloud has become difficult to measure.

On one hand, the availability of advanced technologies via Infrastructure as a Service providers continues to unlock scale and innovation that the traditional on-premises infrastructure model could never support.

At the same time, the broader implications and risks of deploying workloads on publicly hosted infrastructure are increasingly cannibalizing the value that enterprises generate via IaaS.

Today, enterprise adoption of cloud infrastructure has become a high-stakes balancing act between velocity and risk. The more that an organization prioritizes one, the more it seems to sacrifice on the other.

This whitepaper explores these challenges, walks through the impact they have on cloud velocity, and highlights the role that local cloud development plays in helping to increase the overall value of cloud infrastructure.

Cloud Development Challenges in the Enterprise

While public cloud infrastructure adoption continues to grow, enterprises are finding that the true cost to support innovation in the cloud are higher than they anticipated.

Here is a look at the challenges for organizations that rely on public cloud infrastructure for non-production workloads.

Security and Compliance Risks

According to IBM, the average data breach cost enterprises $4.4M in 2024[1]. In the US, that figure surged 9% year-over-year in 2024 to reach an average of $10.2M per breach. Meanwhile, roughly one-third (31%) of data breaches stem from cloud misconfigurations, according to the 2024 Thales Cloud Security Study[2].

In addition to managing cloud security posture across large engineering organizations, enterprises are tasked with maintaining compliance with regulatory standards for data protection. IBM's 2025 Cost of a Data Breach report found that 53% of cybersecurity attacks target customer Personally Identifiable Information (PII), while 50% of respondents to the Thales 2024 Cloud Security Study said that compliance and privacy are more difficult to maintain in the cloud than on-premises.

The continued proliferation of Artificial Intelligence (AI) could compound these challenges.

By 2028, 75% of enterprise software engineers are expected to use AI coding assistants, according to Gartner[3]. The accelerated delivery and increased volume of new code could drive a spike in security and compliance vulnerabilities on live public cloud infrastructure.

Meanwhile, malicious actors are embracing AI equally rapidly to develop more sophisticated cyber attacks, such as those capable of identifying misconfigured cloud resources at scale.

This heightened risk also adds to the cost of operating in the cloud. Every engineer's access needs to be configured with permissions in mind, while every deployment needs to prevent the misconfigurations that can lead to data breaches or regulatory violations.

1. IBM Cost of a Data Breach Report, 2025
2. Thales, 2024 Thales Cloud Security Study
3. Gartner

Cloud Costs and Financial Risks

The consumption nature of public cloud infrastructure billing forces enterprises to justify every hour of every environment their teams deploy.

In 2025, the Flexera 2025 State of the Cloud Report[1] found that 27% of total cloud costs are considered waste, which means engineering and FinOps organizations still have work to do to optimize cloud spend.

The decentralized nature of the cloud also creates a financial risk that many engineering teams need to account for. Gartner has estimated that 69% of enterprises have exceeded cloud budgets[2], while the Flexera survey found that estimates that the average overrun amounted to 15% of the organization's cloud budget.

Similarly to security and compliance, efforts to optimize costs and mitigate budget risk have created additional bottlenecks in day-to-day cloud development, particularly for non-production environments. Idle and over-sized resources provisioned to support tests inflate costs on the cloud bill.

To prevent this waste, many organizations have centralized access to provision cloud environments among engineers equipped to ensure cloud environments adhere to internal standards on cloud tagging, instance sizing, operational runtimes, and other key details affecting the costs on the bill.

As a result, the added time and resources spent to optimize cloud infrastructure end up cannibalizing the efficiency gains that they accomplish.

1. Flexera 2025 State of the Cloud Report
2. Gartner, 2024 Cloud Spending: IT Balances Costs with GenAI Innovations

Productivity and Velocity Challenges

At scale, centralized cloud access has created widespread bottlenecks among the teams that require access to development and testing environments.

In a survey of more than 500 software developers and platform engineers conducted by Rafay Systems, environment provisioning was the leading obstacle to accelerating application deployments, as reported by 61% of respondents[1].

Many enterprises circumvent these challenges by distributing access to shared development environments.

While this helps to streamline both the cost and accessibility of cloud development environments, the shared nature of these environments carries productivity implications of its own. Bugs due to conflicting code updates or other activity among the various developers operating in a shared environment can lead to lengthy downtime that throws off the work for everyone who relies on it.

Meanwhile, shared development environments are still susceptible to the inherent delays of pushing code and debugging infrastructure on publicly hosted infrastructure, which can add up at scale and lead to hours of daily productivity losses for highly paid engineers.

When looking across the cloud infrastructure landscape for a modern enterprise, it's clear that:

- Significant time is spent to govern infrastructure access, configurations, and deployments for security, compliance and cost.
- When developers gain access to development environments, they face delays and downtime due to the nature of deploying and debugging development environments on public cloud infrastructure.
- Enterprises still suffer from the wasted costs and security vulnerabilities that they aim to prevent by centralizing access.

1. Rafay, Environment Provisioning for Application Development & Delivery Survey

Introducing Local Cloud Development via LocalStack

Local cloud development moves non-production workloads to a safe layer of the cloud development tech stack that removes the risk and friction associated with operating on public cloud infrastructure.

By replicating the behavior of fully integrated cloud services on local infrastructure, local cloud development decouples the day-to-day coding and testing process from the public cloud.

This directly supports three pillars of modern cloud-native software engineering:

1. **Security & Compliance:** Engineers reduce their reliance on highly privileged cloud credentials and direct access to staging or production environments during routine development, significantly lowering the potential attack surface for sensitive infrastructure and data. This allows engineers to experiment with cloud technologies without ever being granted access to a cloud account, while also surfacing misconfigurations or other human error before it's deployed on live infrastructure.
2. **Cost Efficiency:** Development and early-stage testing, which account for a large portion of cloud usage, are shifted off expensive public cloud resources (like managed databases and compute instances), minimizing cloud bills.
3. **Productivity:** By removing the security and financial implications of deploying to public cloud infrastructure, local cloud development can help to democratize access to development workloads to more engineers. Additionally, eliminating the latency of remote deployment provides developers rapid, near-instant feedback loops to accelerate debugging, iteration, and experimentation without constant network dependence.

LocalStack began in 2016 as an open-source project designed to address the slow, costly, and complex feedback loops inherent in cloud development. Its core idea was radical: to enable developers to run a functional AWS environment directly on their laptops using a single container.

As more large enterprises began to recognize the valuable role that local cloud development plays in their tech stack, LocalStack evolved to become a high-fidelity emulator supporting some of the most popular services in use among AWS customers, as well as key functionality for data engineering teams that rely on Snowflake.

Today, LocalStack is trusted by hundreds of enterprises worldwide, including some of the world's largest and most innovative financial institutions, software vendors, and media companies.

How does LocalStack work?

LocalStack does not function as an intermediate deployment gateway; rather, it is a local target for standard AWS development tools.

The process of pushing code from LocalStack to the public cloud hinges on the tool's high API compatibility. During local testing, developers configure their existing Infrastructure-as-Code (IaC) frameworks—like AWS CDK, Terraform, or CloudFormation—to point to the LocalStack endpoint running on their machine. This step allows them to provision and test their complete architecture, from S3 buckets to Lambda functions, entirely offline.

Once the application and its infrastructure dependencies are fully validated in the LocalStack sandbox, the deployment shifts to the public cloud by simply redirecting the target endpoint. The developer switches the configuration of their IaC tools or CI/CD pipeline to use the real AWS credentials and official endpoints.

Because the deployment artifacts and IaC templates have already been proven functional against the high-fidelity LocalStack environment, the final deployment to the production AWS account becomes a confident, low-risk process that bypasses the need for extensive, time-consuming cloud-based integration testing.

Large enterprises often scale CI/CD by running LocalStack on clusters, particularly Kubernetes (K8s), as a shared service for automated integration testing. This clustered deployment allows for high-throughput, parallel testing across numerous feature branches, ensuring that entire infrastructure stacks are tested automatically without developers having to maintain a complex local setup or incur cloud testing costs.

Conclusion

LocalStack is a critical component of the modern enterprise GTM strategy and tech stack. It transitions the focus from simply consuming the cloud to intelligently and efficiently developing for the cloud. By implementing secure, local cloud environments, organizations can concurrently mitigate security/compliance risk, optimize costs for non-production workloads, and improve cloud ROI. Ultimately, LocalStack allows enterprises to unlock the full potential of cloud innovation while maintaining control and predictability.

LocalStack

Request a Demo Session

Speak to a LocalStack expert & learn how we can help your organization boost its cloud development:

- Get all your questions answered
- Do a live walkthrough of the app
- See customized extension
- Explore which pricing plan is best for you examples

Get in Touch