



WHITE PAPER

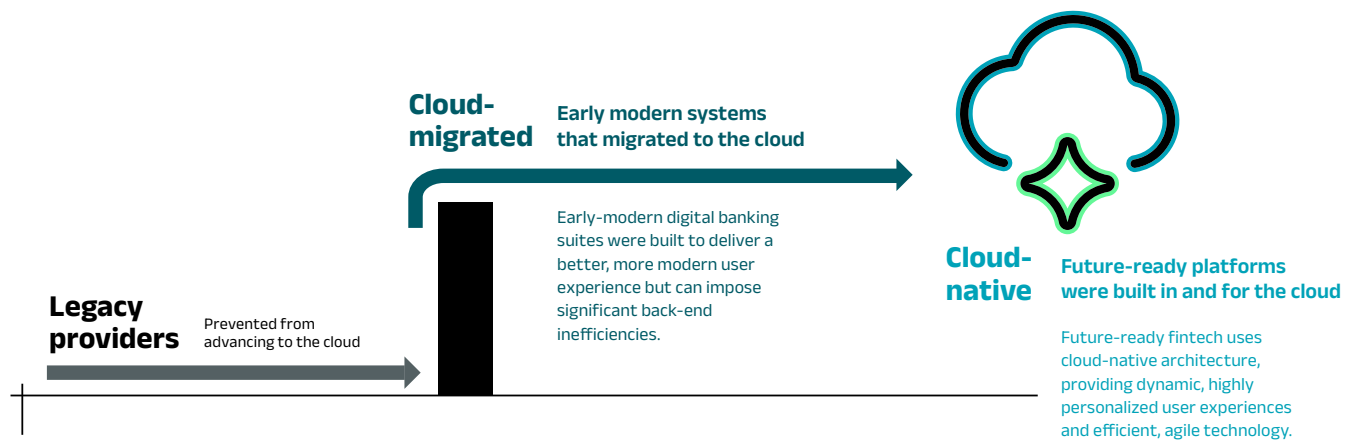
Cloud-native or cloud-migrated: why it matters to the future of your digital banking strategy



Approaches to cloud app adoption

As cloud computing becomes essential to modern financial operations, understanding the nuances between different types of cloud solutions is critical. Two primary approaches to cloud adoption are cloud-native and cloud-migrated applications. Both offer ways to leverage cloud technology, but distinct differences impact scalability, performance, and agility. Knowing these differences can help businesses make more informed decisions, tailoring their cloud strategies to achieve maximum benefit.

Defining cloud-native and cloud-migrated applications



What is a cloud-native application?

Cloud-native applications are designed and built specifically for cloud environments, often termed “born in the cloud.” These applications are engineered to fully harness the capabilities of the cloud, such as scalability, flexibility, and resilience. They often follow modern software architecture practices like microservices and containerization, making them highly adaptable to rapid changes in business needs. In addition, there is a rich landscape of open-source tooling and products that are specifically designed to work with cloud-native applications. Over the long term, this module can yield cost savings as organizations no longer need to maintain physical hardware and the staff required to support it.

- **Microservices architecture**

A key characteristic of cloud-native applications is that, this approach breaks down a large application into smaller, independently deployable services. Each service can be scaled and updated independently with demand spikes without impacting the entire architecture; this flexibility allows for easier maintenance and faster innovation.

- **Immutable infrastructure**

The infrastructure hosting the application is never manually changed. If a server hosting the application requires a change, such as more resources, the old server is replaced with a new server in an automated fashion. This makes updates more predictable and avoids manual mistakes.

- **DevOps culture**

To fully take advantage of cloud-native applications, organizations are more likely to adopt DevOps practices, emphasizing concepts such as continuous integration and delivery (CI/CD) and enhanced collaboration between development and operations teams.

Success for cloud-native solutions depends on significant investment in cloud-savvy talent and creating the right kind of culture. The initial setup can be complex, and much learning is required about new technologies, architectural patterns, and organizational shifts.

What is a cloud-migrated application?

Cloud-migrated applications were initially developed for traditional, on-premises infrastructure. These applications are later migrated to a cloud environment, often without rewriting them to take full advantage of cloud-native features. This strategy allows companies to leverage their existing legacy systems while transitioning to the cloud, gaining some cloud benefits without entirely redesigning their infrastructure. The lower initial investment makes this approach attractive for organizations that lack the resources or readiness to adopt a fully cloud-native solution. Migrating to the cloud incrementally provides a cost-effective path to modernizing IT environments while retaining legacy assets.

Types of cloud migrations

- **Retain/Relocate/Rehost**

Legacy applications are moved into the cloud without making any application changes. These are the easiest types of migration; however, they provide the least benefits compared to a cloud-native application.

- **Replatform/Repurchase/Refactor**

Legacy applications are refactored to work with cloud services. These migrations are more complex and costly. In some cases, the cost of refactoring a legacy application exceeds the benefits gained from doing so.

In both cases, cloud-migrated applications may still rely on legacy systems, infrastructure, and architectural patterns, reducing their benefits compared to cloud-native applications and increasing operational costs. Cloud-migrated applications may also be limited in the cloud services that they can utilize, which can impact scalability and reliability.

	Legacy digital banking products	Early modern digital banking suites (cloud migrated)	Future-ready banking platform (cloud native)
Technology agility, service adoption, and innovation	Outdated infrastructure leads to high maintenance costs, complexity, and slow adaptation to customer needs. Large-scale updates take multiple quarters to start and introduce risks.	While more flexible than legacy systems, early modern digital suites struggle with scalability and rapid innovation. Integration requires extensive effort, leading to slow feature rollouts and growing technical debt.	Cloud-native platforms, designed for agility, enable rapid adaptation with minimal resource investment. Continuous, iterative feature releases ensure quick responses to evolving needs.
Technology budget allocation	A significant portion of IT budgets are spent on maintaining existing systems, system reconfiguration, and operational costs, leaving little room for innovation.	Updates often require infrastructure refactoring, increasing costs before deploying new solutions. Budget constraints slow the adoption of emerging technologies.	Cloud-native design optimizes spend by reducing maintenance costs and enabling efficient allocation of funds toward high-priority innovations.
Customer engagement & personalization	Basic, standardized offerings with limited data-driven, actionable insights. Engagement tools are outdated and difficult to modify, limiting their effectiveness in meeting evolving customer expectations.	Personalization tools exist but are burdensome to deploy and update.	Engagement solutions are seamlessly integrated and easy to update/optimize throughout the user journey, leveraging real-time data and AI to deliver a dynamic, personalized digital experience that enhances financial outcomes and increases user retention.
Scalability of features & enhancements	Even minor changes require extensive planning and execution cycles (min. 4-6 months), slowing innovation and responsiveness.	Built-in silos prevent efficient scalability. Development efforts for small updates still require months (3-4 months) of planning and execution.	Cloud-native microservices allow for rapid, continuous updates (weekly or bi-weekly), eliminating long development cycles while ensuring stability.
Reliability & uptime	Susceptible to failures due to outdated infrastructure, security vulnerabilities, and even power outages. Demand spikes can crash entire systems.	System updates often require forced, planned and unplanned downtime, impacting service availability. Interdependencies between services increase the risk of widespread outages.	Near 100% uptime with independent service updates. A modular approach ensures that system failures remain isolated, minimizing disruptions.
Security & compliance	Keeping legacy systems updated with security patches is complex, time-consuming, and often incomplete, leaving institutions vulnerable.	Cloud migrations introduce risks such as misconfigurations, data blind spots, and multi-tenant vulnerabilities, potentially leading to breaches.	Security is integrated from the foundation, with single-tenant hosting ensuring complete data isolation. Microservices architecture prevents unauthorized access or breaches with other applications.
Extensibility & integrations	Lack of flexibility and well-documented APIs leads to high costs and long procurement cycles for system extensions. New integrations require significant custom development.	Cloud-migrated systems require complex modifications and large-scale upgrades to accommodate new extensions, often leading to unexpected performance issues.	Future-ready, cloud-native fintech solutions provide seamless extensibility with well-documented SDKs and integrated third-party fintech partnerships, allowing for easy customization without heavy development efforts.

The importance of being cloud-native for digital banking

As financial institutions (FIs) increasingly embrace digital transformation, cloud-native technology will become a cornerstone of modern banking. This approach not only enhances operational efficiency but also supports the industry's fast-paced demands. Here's why cloud-native is particularly important for digital banking, benefiting both financial institutions and their users.

Three significant benefits

1. Fast and seamless updates

Cloud-native architecture is better suited for continuous delivery and integration (CI/CD), allowing for frequent updates without downtime. FIs can quickly roll out new features and fix issues.

2. Scalability and flexibility

Cloud-native solutions allow FIs to scale their services up or down based on demand. This flexibility is essential for handling peaks in transaction volumes, such as during holidays or special promotions, without overburdening the infrastructure or the teams managing it.

3. Less downtime

Cloud-native applications are more likely to be designed for fault tolerance and leveraging highly available cloud services. This translates to less downtime and better mean time to resolution (MTTR) when issues occur.

Cloud-native vs. cloud-migrated applications: Key differences for financial institutions

Cloud-native applications are purpose-built to fully utilize the capabilities of cloud computing, offering greater scalability, agility, and security. In contrast, cloud-migrated applications often retain legacy limitations, hindering the ability to fully capitalize on the cloud's benefits.

Choosing a cloud-native approach is essential for FIs looking to stay competitive, agile, and responsive to the digital banking landscape's ever-evolving needs.

This approach ensures that the institution and its users both experience the highest levels of performance, security, and innovation.

