

INTRODUCTION TO CLOUD NATIVE MIGRATIONS



Legacy applications are designed with a specific set of technology stack—and therefore only work well with specific pre-provisioned hardware and software platforms. These monolithic apps cannot be deployed and scaled in separated components in [multi-cloud environments](#), especially considering the underlying technology stack is tightly coupled, highly dependent on traditional hardware platforms, and offers limited integration with the modern cloud stack.

In order to take advantage of cloud computing, legacy apps are migrated to the cloud using the lift and shift strategy: reconfiguring the application and data assets to run on the cloud. However, the traditional cloud migration strategy fails to exploit cloud native features and functions. The cost and complexity of refactoring the app may outweigh the values offered by cloud computing.

Cloud native overview

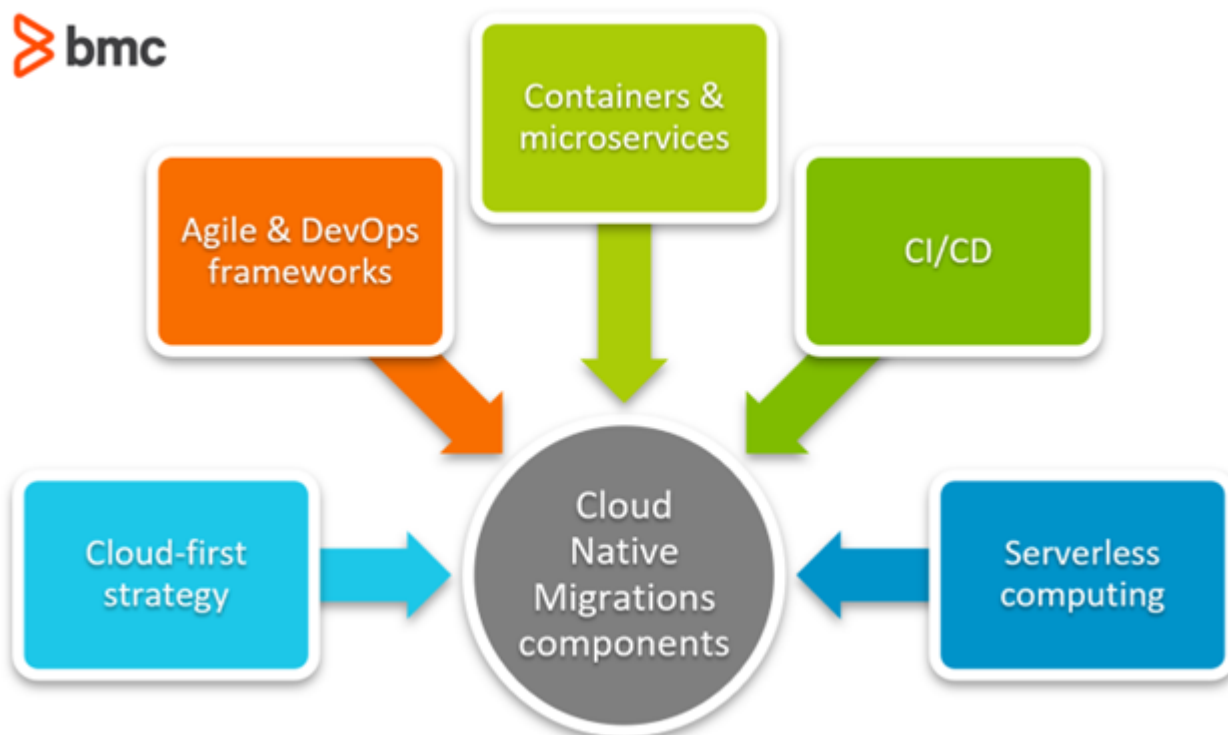
Cloud native applications use open source and [microservices-based](#) technology stacks to build scalable app components in containerized environments. The apps can be managed dynamically using technologies such as Kubernetes and deployed to an existing [Infrastructure as a Service](#) (IaaS) cloud platform.

The idea is to develop apps that are robust and cloud agnostic, allowing organizations to leverage pre-built cloud-native features via [application programming interfaces](#) (APIs). Of course, the apps written for a specific cloud service are portable only to alternative platforms where the same APIs exist. Organizations are therefore required to assess the tradeoffs—cloud performance and cost

advantages—against [vendor lock-in](#) and integration across cloud platforms.

Components of cloud native migrations

Let's review some of the key elements of cloud native applications and discuss the corresponding industry trends according to the Cloud Native Computing Foundation 2020 [research report](#) that drew responses from enterprises and SMB firms around the world:



Cloud-first strategy

A cloud-first strategy refers to the practice of developing a service-oriented organization focused on the specific capabilities most relevant to the market. The idea is to push and scale app features following fast release cycles, especially since market dynamics remain unpredictable and ever evolving.

Survey highlights:

- [Public cloud](#) is the most popular model of choice for 64% of the respondents.
- Around one-third deploy hybrid and multi-cloud environments in some capacity.
- On-site and private cloud deployments remain popular for legacy monolithic apps for over half of the responding organizations.

(Read more about [software & app modernization](#).)

Agile & DevOps frameworks

Organizations must follow rapid release cycles to take full advantage of cloud-native features—concepts that are central to [agile and DevOps ways of working](#). The ability to provision and scale resources on a whim makes it easier for organizations of all sizes to compete with large enterprises, while only paying on a usage-based service model.

Survey highlights:

- The number of organizations releasing app components and updates multiple times a day reached 29%.
- Over half of the responding organizations release software feature updates at least on a weekly basis.
- Advanced IaaS features have been the key to achieve fast release cycles and exploiting cloud-native architectural design patterns.

Containerization & microservices

Use of container services in production is the norm for [DevOps teams](#) migrating existing apps and services to the cloud-native architecture. [Kubernetes](#) has emerged as the leading technology platform for cloud-native apps and the ecosystem surrounding the technology—including community support, certifications, and open-source tools—is growing rapidly.

Survey highlights:

- 91% of the respondents use Kubernetes for cloud-native applications.
- Thousands of open-source developers offer continued support for the platform updates, security, and feature improvements.

(See how [containers & Kubernetes](#) work together.)

CI/CD

As part of the DevOps software development lifecycle (SDLC) framework, engineering teams use [Continuous Integration and Continuous Deployment](#) (CI/CD) practices to automate the process of developing, testing, and deploying app components. CI/CD for cloud-native apps is, simply put, the same CI/CD process that specifically supports the API services used to build and run cloud-native apps.

Survey highlights:

- 82% of the respondents use CI/CD pipelines in the production environment.
- Jenkins (53%), GitLab (36%) and GitHub Actions (20%) are the most popular choice of tools.

(Compare [GitLab to GitHub](#).)

Serverless computing

[Serverless computing](#) is the practice of building and running software without having to manage the underlying [server infrastructure](#). The hardware is decoupled and used as a configurable functionality for containerized apps.

The functionality is managed by [service mesh platforms](#) (such as [Istio](#)) that control how the app components communicate with the hardware and other app components. The service mesh operates as a layer of infrastructure that decouples the underlying hardware.

Survey highlights:

- Around 30% of the organizations use serverless technologies in production and another 35% is

evaluating or considering the technology for use in the next 12 months.

- 60% of serverless technologies are used as a hosted platform.
- [AWS Lambda](#) (57%) and [Google Cloud Functions](#) (27%) are the most popular serverless platforms.
- 18% of the respondents are already using the service mesh technology and another 47% are evaluating it for use in the near future.

Migrating to the cloud

When it's time to migrate to the cloud, BMC is here to help! Our free-to-use tool, [BMC Helix Cloud Migration Simulator](#) makes it easy to:

- Reduce the time, complexity, and costs of migrating servers and workloads to the cloud.
- Improve the accuracy of your planning process by evaluating and right-sizing your workload migration before you select a service provider.

This gives you the insights you need to understand your resource requirements against the cost of maintaining your service levels—and stay within your budget. The BMC Helix Cloud Migration Simulator allows you to compare the costs and resource requirements of leading public cloud providers like Amazon Web Services (AWS), Microsoft Azure, Google Cloud, and IBM Cloud—and it's absolutely free.

Related reading

- [BMC DevOps Blog](#)
- [Cloud Native Security: A Beginner's Guide](#)
- [What's CNAB? The Cloud Native Application Bundle Explained](#)
- [Getting Started with Cloud Native Applications](#)
- [Containers vs Microservices: What's The Difference?](#)
- [Hadoop vs Kubernetes: Will K8s & Cloud Native End Hadoop?](#)