Capgemini engineering

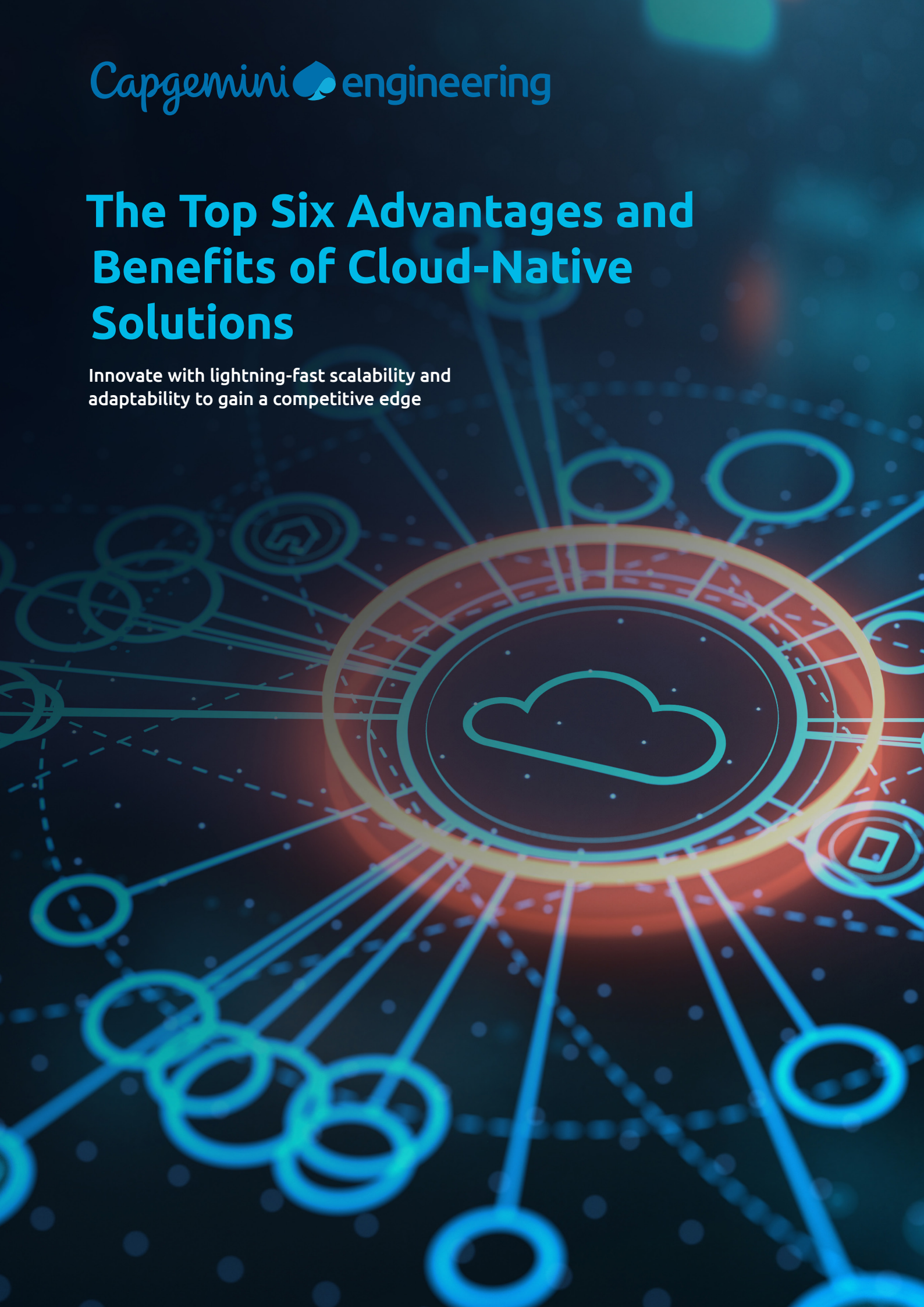# The Top Six Advantages and Benefits of Cloud-Native Solutions

Innovate with lightning-fast scalability and adaptability to gain a competitive edge

Executive Summary

In today's fast-paced digital world, enterprises are challenged to deliver the most innovative, disruptive, and cost-effective solutions for their customers. Through a software-driven approach to products and services, where rapid adoption of modifications is a key premise, enterprises can address these challenges and change the way they design, build, and use software applications.

Cloud native is a software development approach for building, deploying, and improving apps. Its main features are scalability and adaptability delivered through a service-based architecture, microservices, containers, and API-based integration for public, hybrid or private cloud networks. As a consequence, the adoption of cloud-native services is growing rapidly. The public cloud infrastructure market alone is forecast to grow 35% to $120 billion in 2021, with AWS, Azure, and Alibaba leading the way among public cloud providers.[1]

1. Bartoletti, Dave "Predictions 2021: Cloud Computing Powers Pandemic Recovery," Oct. 19, 2020, Forrester
https://go.forrester.com/blogs/predictions-2021-cloud-computing-powers-pandemic-recovery/

Why should you care about cloud native?

Traditional enterprise applications were developed and deployed as monolithic entities, run in either on-premises datacenters or "lifted and shifted" into a cloud environment. Application development considered the source code fragments development and the current deployment effort, whether planning the infrastructure or building deployment pipelines for the artifacts. These monolithic entities were not scalable because they had to be redesigned each time the scale changed, even if they were already running in the cloud. Thus, they did not leverage the scalability and superior flexibility ensured by cloud services.

Even if applications were designed to scale, they were not taking advantage of elasticity when it comes to resource usage, compromising the investment in innovation with new solutions. These innovations, considering a monolithic approach, would take several months to complete. Using a cloud-native approach, the roll-out of a new change would shift from months to a couple of minutes.

Typically, software development had two main lifecycle phases: one before and another after reaching productive environments. With DevOps, development teams are responsible for the entire lifecycle of their products, taking ownership of the solutions they design, build, and deploy. It drives more changes, more quickly with higher stability and quality.

Cloud-native apps are built from the ground up, using Platform-as-a-Service (PaaS) tools with modular architectures, which allow them to scale accordingly as the number of users or business requirements increases. They are composed of many microservices – with each one being fast to develop, deploy, and modify – integrated into today's logic of rapidly changing business needs. Cloud-native apps allow businesses to improve their velocity and agility with flexibility unattainable from monolithic systems with these modular architectures.

The top six advantages and benefits of cloud-natives solutions

Cloud native is not only a paradigm shift in how software is built; it is a cultural change that impacts the entire business

01 Lifecycle management through DevOps
DevOps-driven development ensures faster software delivery and better performance by enabling collaboration between teams and departments in a cycle of rapid, frequent, and reliable software delivery.

03 Collaborative environments through contract-based interaction
The more enterprises invest in API-based integration, the more they will improve their ability to react quickly to new challenges and deploy new solutions for their customers.

05 Self-service, elastic on-cloud deployment
At least 60% of backend developers are now using containers to deploy their solutions. By 2022, 75% of enterprises are expected to run containers in production.

02 Customer-centric software development
With the single cross-functional autonomous team focused on delivering hign-quality services that run isolated through API-based integration, the business can focus on develping client features that generate revenue.

04 Higher complexity managed through serverless platforms
There are now over 6.5 million developers working with cloud-native technologies, four million of them using serverless architectures and cloud functions.

06 Enhanced architectures for improved resilience
Whether it is a preferred base language or a diverse technological stack, enterprises should use API gateways and services meshes to guarantee security and communications throughout the microservices architecture.

Cloud-native benefits

1. Lifecycle management through DevOps

The faster an organization can conceive, build, and distribute a solution to its customer, the more likely it is to avoid disruption in a fast-paced, changing environment. Time-to-market has become the critical differentiator between the most innovative organizations and the ones left behind. Modern DevOps implies a much more predictable and faster transformation of business processes, where automation is the key. The software delivery process has gone through a vast shift based on automation, whether automated testing or automated deployment.

The DevOps market is growing fast

In 2019, the DevOps market exceeded $4 billion and is forecast to grow over 20% a year between 2020 and 2026, according to Global Market Insights.[2] A single team composed of cross-functional members working collaboratively is crucial to delivering maximum speed, functionality, and innovation. The team uses an automated process for testing both delivery and deployment to ensure frequent high-quality deliveries. This is a significant cultural change requiring collaboration between development, operations, product management, and business stakeholders. (See Figure 1.)

According to the DORA DevOps Research Group, DevOps-driven development ensures faster software delivery and better performance by enabling collaboration between teams and departments in a cycle of rapid, frequent, and reliable software delivery.[3] Specifically, the DORA study found that elite DevOps performers deploy 208 times more frequently and have lead times 106 times faster than low performers. In addition, they are twice as likely to meet or exceed their organizational performance goals.
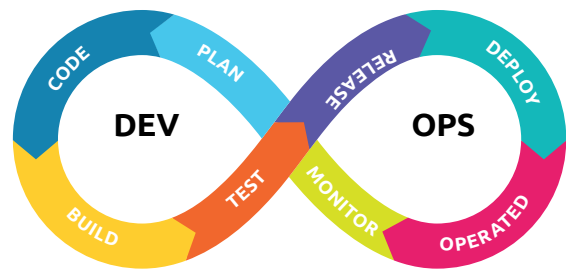

Figure 1. DevOps relies on a single team composed of cross-functional members with responsibility for the component lifecycle

2. "DevOps Market Size by Component," Apr. 2020, Global Market Insights https://www.gminsights.com/industry-analysis/devops-market

3. Wright, Tori, "DevOps can elevate performance, but only if it's done right," Oct. 7, 2019, Diginomica https://diginomica.com/devops-accelerating-do-it-right

2. Customer-centric software development

A cloud-native app not only supports an automated process but leverages it through new designs and standards aiming for portability and rapid modifications. Building and shipping new features faster and iterating continuously requires a great customer experience alongside a mobile-first approach to software development with human-centric design practices, especially when the unique mobile user market is 5.22 billion and counting.[4] (See Figure 2.)
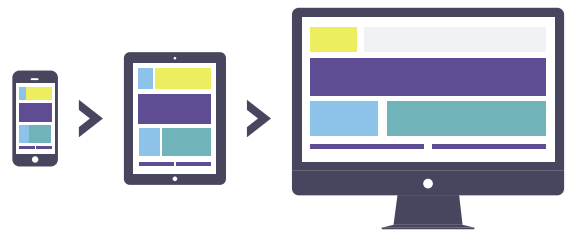

Figure 2. Mobile-first software development

With the single cross-functional autonomous team focused on delivering high-quality services that run isolated through API-based integration built on top of a microservices approach, the business can focus on developing client features that generate revenue. Making quick modifications on a single service that don't affect service availability and are iterating continuously are critical drivers for building experiences that focus on what matters most for customers.

4 Kemp, Simon, "Digital 2021: Global Overview Report," Jan. 27,2021, Data Report https://datareportal.com/reports/digital-2021-global-overview-report

3. Collaborative environments through contract-based interaction

An enterprise's capacity to leverage integration inside and outside the organization through API-based communication is crucial to building better customer experiences on top of data-ready applications and robust but clean frontend applications.

The APIs that are used leverage both internal and external service usability when defined as contracts between the data producer and the consumer and can add extra unforeseen value for businesses. Those APIs should establish a contract between the endpoints, allowing for each of them to be in totally different technological stacks or programming languages. The standardization of communications allows a single service-specific endpoint to be redesigned and rebuilt for a more generic purpose.

The more enterprises invest in API-based integration, the more they will improve their ability to react quickly to new challenges and deploy new solutions for their customers. For example, Salesforce.com generates 50% of its revenue from API integrations. "APIs are the windows to new ecosystems [that] allow firms to expand into markets they may never have previously considered."[5]

4. Higher complexity managed through serverless platforms

Cloud-native applications make infrastructure management effortless using PaaS platforms. They are now expanding into serverless platforms that allow you to upload portions of code in the form of functions. (See Figure 3.) There is virtually no effort with instantiating or configuring the network infrastructure, nor with deploying or allocating new computational resources. In fact, according to the Cloud Native Computing Foundation, there are now over 6.5 million developers working with cloud-native technologies, four million of them using serverless architectures and cloud functions.[6] These applications aim to run on containers, remove the need to manage the infrastructure, accelerate application development speed, and be invoked and scaled individually.

Serverless should be used when the specific service functions do their logic isolated without migrating an entire existing application. Each service function has its own logic for completing the request, without the logic being orchestrated by the central server application like in the original versions. This gives each service function a more architecturally aware role.
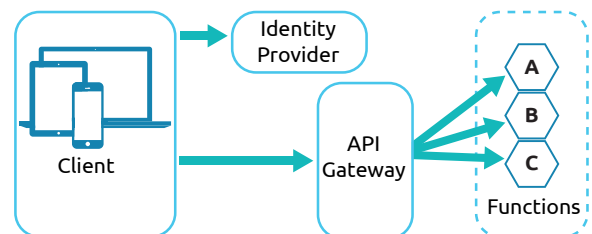

Figure 3. Serverless architecture

5. Iyer, Bala and Subramaniam, Mohan "The Strategic Value of APIs," Jan. 7, 2015, Harvard Business Review https://hbr.org/2015/01/the-strategic-value-of-apis
6. The State of Cloud Native Development," Aug. 14, 2020, Developer Economics, SlashData, and Cloud Native Computing Foundation
https://www.cncf.io/wp-content/uploads/2020/08/CNCF-The-State-of-Cloud-Native-Development_Q419.pdf

5. Self-service, elastic on-cloud deployment

It is possible to manage and secure cloud-native applications independently of the hosting infrastructure that supports them by using containers managed at scale in the cloud. Reducing enterprise costs is possible with a long list of powerful open-source tools running serverless, dynamic workloads, and pay-per-use computation and network resources. (See Figure 4.) At least 60% of backend developers are now using containers to deploy their solutions.[7] By 2022, 75% of enterprises are expected to run containers in production.[8]
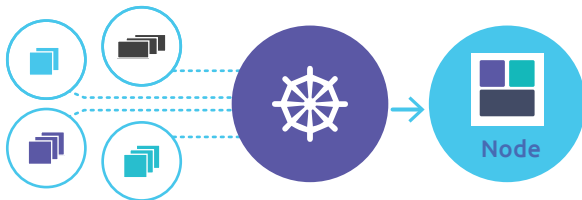

Figure 4. Kubernetes automatically deploys, scales, and manages containers

Containerized solutions enable more efficient use of resources, as they are agnostic about the hosting platform, ensuring that dependencies are met. With containers, developers can enhance the business services that the solutions implement, which offers a separation of concerns between infrastructure and resources management.

Containers make the same provisioning considerations, which container orchestration mechanisms can handle. However, monitoring and compliance should also be considered. Is the team set up suitably for a DevOps approach that can take full ownership of the solution, from design to production monitoring? Is the containerized solution compliant with regulatory standards and rules, or are they handling personal data? These are among the questions to address about software design and development, but the infrastructure should also be considered.

The elasticity provided by cloud-native development is especially beneficial when services have unpredictable traffic peaks and resource requirements that are not foreseen or planned upfront. It is a self-service logic where enterprises pay for resource usage rather than resource allocation.

6. Enhanced architectures for improved resilience

Cloud-native applications aim for better reliability, as they have self-healing mechanisms based on automation, with fault-tolerant resilient architectures mainly driven by a microservices approach. With the business process divided into multiple microservices, each acting as a service function, errors can be tracked and isolated to the service creating the error. This trace will fix the problem at its origin, without significant service downtime, like what happened in the era of monolithic architectures.

Microservices architectures are designed on top of the domain-driven design that decouples the domain logic of the service, so each microservice domain is segregated, achieving loosely coupled, independent microservices. The communication is done asynchronously by REST endpoints and message queues and accessed by micro-frontends.

Whether it is a preferred base language or a diverse technological stack, enterprises should use API gateways and service meshes to guarantee security and communications throughout the microservices architecture. (See Figure 5.) One of the challenges is observability. There is no centralized log built by a single monolithic application, but rather several microservices that interact to fulfill specific request choreographies.
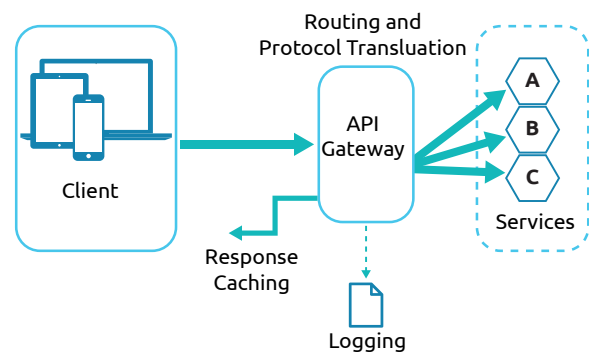

Figure 5. API gateway benefits

7. Ibid. https://www.cncf.io/wp-content/uploads/2020/08/CNCF-The-State-of-Cloud-Native-Development_Q419.pdf

8 . Moore, Susan "Gartner Forecasts Strong Revenue Growth for Global Container Management Software and Services Through 2024," Jun. 25, 2020, Gartner https://www.gartner.com/en/newsroom/press-releases/2020-06-25-gartner-forecasts-strong-revenue-growth-for-global-co

Conclusion

The cloud-native mindset will not be realized with a subset of modified practices or just another new tool placed in the enterprise's production environments. Cloud native is an entirely new set of methodologies, principles and management rules that will help you achieve true business innovation. Without this game-changer of a business strategy, enterprises looking to disrupt markets will have a tough time.

Whether you are considering renewing the application portfolio or modernizing it, enterprises should embrace cloud native to ensure they win competitive battles in rapidly changing markets.

Cloud native is not only a paradigm shift in how software is built; it is a cultural change that impacts the entire business. Cloud native is an innovation mindset that combines organizational and technical changes in the way software is designed, built, and deployed. It is a set of cultural, organizational, and technological enhancements to software engineering that leverage innovation, collaboration, and agile methodologies. This new mindset will bring more frequent and shorter delivery cycles and higher delivery quality. It will establish new and higher expectations for business innovation and competitive advantage.

Author

Pedro Bonifácio Costa
Advanced Architect
Capgemini Engineering

Pedro is an experienced software architect, well versed in solution architectures that are built to last, with a keen interest in topics ranging from data-ready apps to cloud-native development. He earned a BSc in Information Technology and an MSc in Open Source Development and has been working in the IT world for the last 20 years.

About Capgemini Engineering

Capgemini Engineering combines, under one brand, a unique set of strengths from across the Capgemini Group: the world leading engineering and R&D services of Altran - acquired by Capgemini in 2020 - and Capgemini's digital manufacturing expertise. With broad industry knowledge and cutting edge technologies in digital and software, Capgemini Engineering supports the convergence of the physical and digital worlds. It helps its clients unleash the potential of R&D, a key component of accelerating their journey towards Intelligent Industry. Capgemini Engineering has more than 52,000 engineer and scientist team members in over 30 countries across sectors including aeronautics, space and defense, automotive, railway, communications, energy, life sciences, semiconductors, software & internet and consumer products.

Learn more at:
www.capgemini-engineering.com

Write to us at:
engineering@capgemini.com