ISSN: 2754-6705

Journal of Mathematical & Computer Applications

SCIENTIFIC Research and Community

Review Article

Open Access

Transforming Application Development and Deployment using Serverless Computing

Rajesh Kotha

Software Development Engineering Advisor at Fiserv, USA

ABSTRACT

Serverless computing which is sometimes known as "Function as a Service" is a digital computing and data system that functions without traditional servers (FaaS). This research study explores the design, advantages, uses, and impacts of serverless computing. The introduction explains the fundamental ideas of serverless computing. FaaS is one of its distinguishing characteristics. Also, it investigates and expounds on the effects of serverless computing on flexibility, the creation of applications, and the ecosystem of cloud computing as a whole. This offers a chance to investigate the emerging trend of serverless technology and how it affects and improves current cloud-based application development techniques. This study also explores how serverless computing is changing how organizations deploy and operate cloud-based applications, as well as how it affects conventional cloud services. Real-world case studies are included to better illustrate the success of FaaS. As a depiction of modern cloud technology, serverless computing highlights the evolution of cloud programming designs, concepts, and platforms as well as the breadth and depth of cloud technology development. Serverless computing makes development faster because the deployment process becomes much simpler.

***Corresponding author**

Software Development Engineering Advisor at Fiserv, USA.

Received: June 09, 2022; **Accepted:** June 16, 2022, **Published:** June 23, 2022

Keywords: Serverless Computing, Function as a Service (FaaS), Event-Driven Architecture, Serverless Architecture, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Cold Start Latency, Serverless AI Applications, Scalability, Serverless Framework

Introduction

Cloud computing is increasingly becoming popular across the globe [1]. Cloud services are accessible and usable through internet-based platforms. The advantages that cloud computing offers, like reliability, scaling, and versatility, have led to software developers incorporating it into their products. "In general, cloud computing provides three services: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS). Part of the serverless computing concept is Function as a Service (FaaS)" [2]. Although the terms "serverless" and "FaaS" are occasionally used interchangeably, the former refers to the functionality execution in a serverless framework.

Multiple crucial components work in tandem to form a serverless architecture (Figure. 1). At its center are serverless functions, which are small, lightweight code snippets that perform actions in reaction to events. These routines are executed as a result of such events, which might include schedules, updates to the database, or requests made via HTTP. An API gateway provides capabilities like authorization and rate limiting while acting as a gateway site for external requests, forwarding them to the appropriate services. The cloud provider assigns resources to carry out tasks and scales the computer infrastructure dynamically.

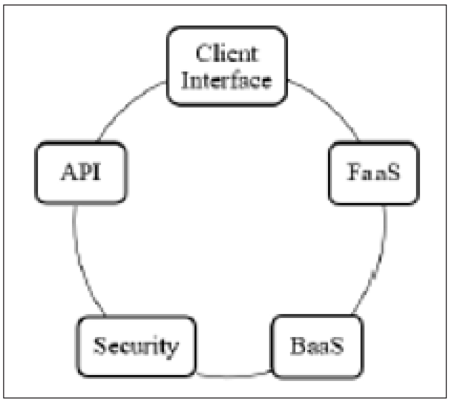

Figure 1: Components of Serverless Architecture

The transition from on-premises infrastructure to serverless cloud-based solutions suggests a dramatic change. This evolution entails several innovative changes that will completely change how businesses and software developers approach their technological environments. Conventional infrastructure supervision involves several operational tasks, such as setting up servers performing repairs, and monitoring them (Figure. 2). These responsibilities are transferred to the cloud provider via serverless cloud systems [3]. As a result, developers can focus on tasks that bring value rather than infrastructure issues because of this laissez-faire management approach. Digital market, enterprises can remain ahead of the competition, produce new products faster, and drive company success by utilizing the advantages of serverless computing.

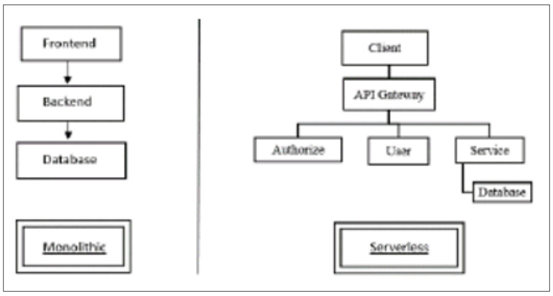

Figure 2: Traditional Infrastructure Vs. Serverless Cloud

Scope

This study gathered data from a variety of sources to precisely address important research issues about serverless cloud computing. This will aid in the understanding of serverless cloud computing and enable academics and developers to make valuable contributions to its advancement. These sources specifically explore the design, advantages, uses, and impacts of computing to developers while other studies concentrated on technical interpretations of the technology. These days, there are several applications for this kind of computing. Therefore, the sources used set out to examine the industry's serverless computing development methods through an empirical investigation. As a result, more resources and easy access to them would encourage developers to make use of serverless computing's potential. The concept of serverless computing has a lot of potential to progressively broaden its applications to include a wider variety of sectors within the context of the IT industry. Consequently, serverless computing is not limited to infrastructure enhancement but may be applied to a multitude of tasks such as training and communications.

Literature Review

Following the emergence of digitalization in both IT and hardware structures, cloud service providers began to use cloud computing more and more to deliver their services to clients. "Based on the way services are provided, cloud computing may be broadly classified into three categories: platform as a service (PaaS), infrastructure as a service (IaaS), and software as a service (SaaS)" [4]. Cloud service management is by no means a simple process. The authors of have discussed several issues that users may encounter when administering a cloud system, including security, auto-scaling, availability, load balancing, and monitoring [2]. For instance, the user of the cloud must guarantee that the services are available and that the failure of one computer does not impact the other machines. To safeguard the services from natural catastrophes, he or she must also think about geographically dispersing duplicates of the offerings. Managing load is an additional problem. In this scenario, it is the cloud user's responsibility to make sure that service demands are balanced to make the best use of all available resources.

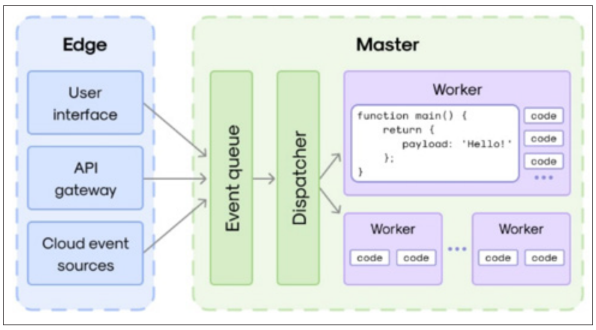

Figure 3: The Functionality of FaaS

A new computing approach known as serverless cloud computing has been introduced as a result of these difficulties. As seen in Figure. 3, "serverless cloud computing provides function as a service (FaaS) and backend as a service (BaaS)" [2]. Services like messaging, storage, user administration, and so forth are included in the BaaS. The BaaS offers key operations like messaging, user authentication, databases, and other things that the FaaS depends on. FaaS, commonly referred to as "event-driven functions," is regarded as the most popular serverless approach.

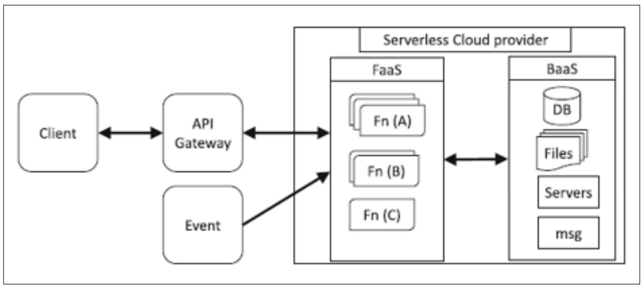

Figure 4: Serverless Architecture [1]

"Amazon Lambda developed and implemented the serverless cloud concept for the first time in 2014. In 2016, cloud businesses like Google and Microsoft embraced it" [4]. In addition to abstracting out of the server-side administration from developers, serverless cloud technology provides another layer of refinement to the current cloud computing models. The serverless architecture frees administration so they can concentrate on the application logic. As an illustration, developers launch their apps as functionalities on the serverless cloud (see Figure. 4). After that, the cloud provider has to administer, scale, and supply various resources to guarantee the seamless operation of these operations. Despite having certain limitations, serverless cloud computing has many advantages; the first one is scalability. In that, both vertical and horizontal scaling are possible through this process; vertical scaling modifies the number of cores in an operating container, whereas horizontal scalability adds or eliminates containers without changing the way resources are currently allocated. Applications with serverless computing dynamically scale up and down in response to user demand, freeing developers from worrying about scalability concerns. An application running in a serverless cloud, for instance, may scale up in response to a rise in requests from the application.

Problem Statement

The maintenance of any kind of server is not an easy task, owing to the substantial expenses involved or the requirement for proficient personnel. These issues are resolved by serverless computing, which offers an affordable means for developers to create and execute cloud applications. “The circumstances that trigger functions are set by configuring event sources” (Figure 5) [2]. In serverless computing, as opposed to conventional cloud computing, whereby connectivity, storage, and an actual or virtual server are set up, the infrastructure is managed by the cloud provider, who also automatically allots resources as the application grows. Put another way, serverless technology is the separation of the server from programmers, freeing them up to concentrate more on the apps they are creating rather than the hosting infrastructure. FAAS is a serverless application that is split up into separate serverless functions that are activated by distinct events from sources like HTTP requests, database modifications, message inquiries, or file uploads. The distribution of computer resources according to demand or an increase in workloads is known as auto-scaling. Companies or programmers may swiftly release applications and receive customer feedback right away with serverless applications. This is crucial for startups since it cuts down on the amount of time and labor required to develop applications.

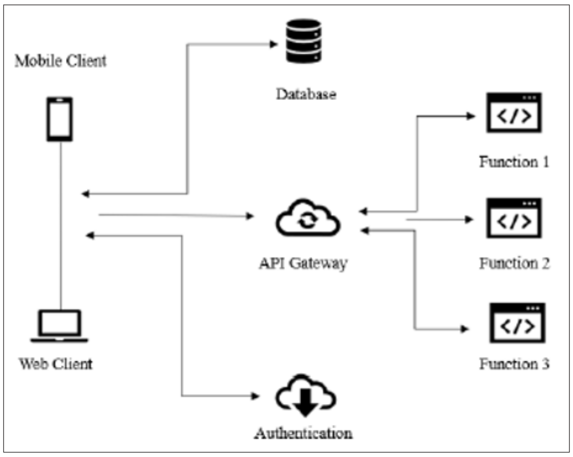

Figure 5: Serverless Application Development

Serverless apps may be used anywhere, in contrast to those hosted on dedicated servers [5]. This enhances application performance and lowers latency. Instead of charging consumers for resources allotted, serverless systems charge users for the resources consumed. With serverless systems, costs are metered, meaning that customers pay only for the number of invocations or the amount of time a function runs. As a result, it is quicker and more effective than conventional methods. If the serverless provider allows users to upload photos to their platform, and the concurrency limit is set to 100 by default, then any requests that exceed this limit are queued and executed during the subsequent function execution. The key characteristic of serverless computing that contributes to related efficiency and flexibility is that resources are not squandered while the server is under low stress or demand, and downtime is avoided when the server is under heavy strain.

Solution

A new computing approach known as serverless cloud computing has been introduced as a result of the difficulties associated with traditional server-based computing. This evolution entails several innovative changes that will completely change and improve how businesses and software developers approach their technological environments. Serverless computing lets the cloud provider handle the infrastructure required to run the code (Figure 6). Developers may use serverless computing to distribute code in more reasonable, smaller chunks called functions. These functions can be activated by many events, including file transfers, changes to databases, and requests made via HTTP. Developers can release code faster and more easily using serverless computing, which shortens the time it takes to provide updates and new features. thus, businesses may get a competitive edge by being able to innovate more quickly and remain one step ahead of the competition thanks to this simplified development process.

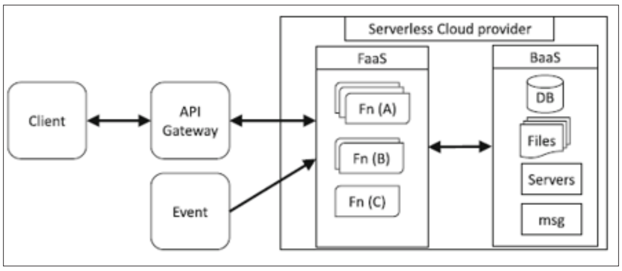

Figure 6: Serverless Architecture [1].

The management, deployment, and maintenance of traditional server-based systems can take a large amount of time, effort, and more importantly, finances (Figure 7) [6]. By using serverless architecture, enterprises may outsource server administration to cloud providers, simplifying their development process and cutting complexity. Because of this simplicity, companies are free to concentrate on refining their goods and providing value to consumers rather than maintaining infrastructure. Moreover, serverless architecture provides more flexibility and agility, allowing companies to swiftly adjust to shifting consumer demands and market situations. Organizations can quickly expand their apps to meet demand, roll out updates without any downtime, and affordably test out new features and functions using serverless computing.

Cost Element	Traditional Server	Serverless Framework
Server Expense	Fixed monthly charge	Pay-per-use
Scaling Expense	High (manual scaling)	Low (automatic scaling)
Operational Expense	High (routine maintenance)	Low (handled by provider)
Hidden Expenditures	Low	Potential for high costs (data transfer, refactoring, training)

Figure 7: Cost Comparison between Traditional Servers and Serverless Frameworks

The cloud operator continuously controls the setup and distribution of servers in a serverless architecture. Applications with unpredictable or unexpected workloads, where streamlining operational administration and cutting expenses are top concerns are ideally suited for serverless architecture. Applications that require a high degree of environmental control and consistent performance might benefit from the use of traditional architecture. As a result, the decision on the computing approach should be based on operational preferences, activity, and specific application needs. Serverless architecture is a recommendable alternative for businesses wishing to cut expenses, boost productivity, and simplify their software development procedures [7]. In today's fast-paced digital market, enterprises can remain ahead of the competition, produce new products faster, and drive company success by utilizing the advantages of serverless computing.

Uses

Serverless computing allows programmers to create and execute application programs without worrying about upkeep duties like safety surveillance, and update installation. Serverless computing has grown in popularity as a technique for businesses trying to offer developers more time to build and release code since the advent of cloud computing. A serverless framework does not imply serverless computing. For instance, a cloud service provider (CSP) manages server administration and other responsibilities in a serverless architecture. Also, serverless technologies can provide on-demand resource provisioning for cloud providers. Billing for serverless computing only occurs when code execution begins and concludes.

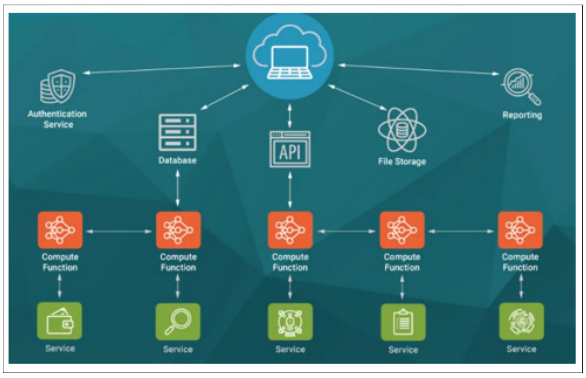

Figure 8: Serverless Infrastructure: (pay as you go) [4].

FaaS, which lets users run code in response to events, is essential in serverless environments to relieve developers of the burden of maintaining the underlying infrastructure. Thus, developers can concentrate only on creating applications. IaaS, is a pay-as-you-go (Figure 8) cloud service that makes virtual machines and other resources available via the Internet. The ability to grow fast is provided by IaaS, which eliminates the need for large, upfront capital costs associated with purchasing, establishing, and maintaining on-premises infrastructure. IaaS increases customers' purchase capacity. The former essentially charges customers for resources up front, whereas serverless billing just accounts for the real-time the code runs. Artificial intelligence (AI) applications have grown significantly in the previous several years. Serverless makes it possible to operate event-driven AI, in which real-time decision-making skills are informed by an ongoing stream of intelligence. Additionally, serverless can assist in resolving the scalability issue that arises when developing new AI, by letting developers concentrate on training rather than the underlying infrastructure.

Companies or programmers may swiftly release applications and receive customer feedback right away with serverless applications. People and businesses now need to access and maintain data better, compared to what they did previously due to the rise of cloud computing. Businesses are depending less and less on physical infrastructure in favor of cloud services to do this. The serverless architecture (Figure 9) is the most recent in a line of coding environment architectures that have been developed to let programmers concentrate more heavily on what they know best—to write and deliver code.

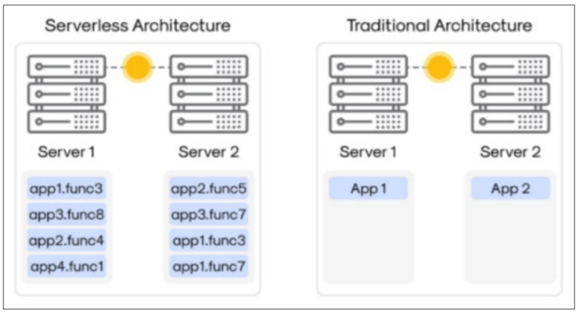

Figure 9: Serverless vs. Traditional Architecture

Impact

Serverless systems bill based on actual use rather than pre-allocated services [8]. Companies only pay for the processing power expended while executing serverless functions. This eliminates the need to provide and pay for resources that are not used, which saves money as seen in (Figure 10). Resources in a serverless system scale upward or downward on their own according to relative requirements. The cloud provider maintains the scalability of the containers while functions are carried out in individual containers. By ensuring that the software can handle fluctuating demand levels without overprovisioning, dynamic scaling helps to minimize resource waste. While handling potential traffic surges, traditional server architectures occasionally need to maintain a specific level of capacity. This leads to the cost of idle resources during times of low demand. Serverless computing on the other hand lowers these idle costs because resources are only used when a function is called upon.

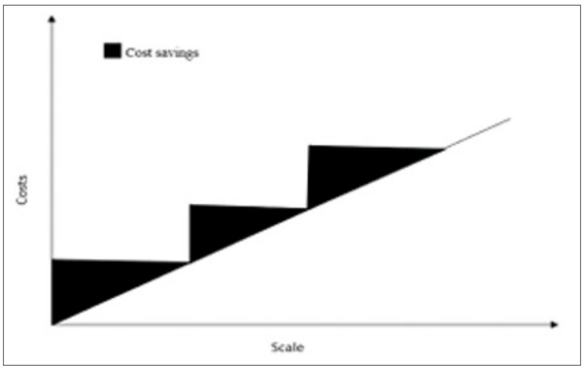

Figure 10: Cost Comparison between Cloud Server Computing and Serverless Cloud Computing

The fact that different settings are used for different functions guarantees that each one gets the resources it needs without interfering with the others. Both overprovisioning and underutilization are prevented by this efficient allocation. Since these operational issues are handled by the cloud provider, developers may focus only on developing the functions. Businesses may easily serve customers worldwide by deploying serverless apps across several areas. Because the cloud provider controls function distribution, global scalability is facilitated.

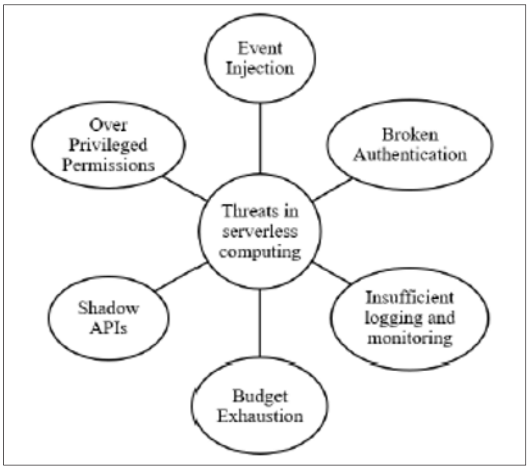

Figure 11: Threats in Serverless Security

Despite being beneficial, serverless computing causes certain challenges (Figure 11). The efficacy and reactivity of serverless operations may be impacted by the cold start latency issue. Anytime a function is activated following a period of inactivity, the cloud service provider is required to supply the resources required for execution. Vendor lock-in arises when serverless computing is used since proprietary offerings link applications to specific cloud providers. Because of disparate APIs, services, and arrangements, this might complicate provider switching or app relocation [9]. Maintaining distributed systems may become complex in serverless architectures due to the presence of many functions, services, and integrations. It is crucial to monitor and troubleshoot these distributed components with great attention to detail.

Conclusion

Serverless computing's architecture and underlying event-driven methodology demonstrate how it has liberated developers from infrastructure administration and resource provisioning. This made it possible for developers to concentrate on producing distinctive features that directly meet end customers' needs. The necessity of meticulous design and strategic planning was highlighted by concerns about serverless cold start delays and vendor lock-in. Besides, the security and compliance elements are more complicated in the setting of dynamic, event-triggered systems, even if they are exclusive to serverless architecture. The rapid development of cloud computing is largely demonstrated by serverless computing, which is consistent with the broader industry trends of automation, and effectiveness. To thrive in the current cloud-driven application development era, companies need to be aware of the subtleties of serverless computing and make intelligent use of its advantages. Consistent development processes, and cold start performance are common challenges associated with serverless computing. Despite that, serverless trend points toward a future where developers are free to express their creativity without having to worry about infrastructure management getting in the way. All things considered, serverless computing represents a paradigm shift that facilitates innovation and progress in cloud computing.

References

1. Sadowsky SJ (2022) Comments regarding: Xu X, Hu B, Xu Y, Liu Q, Ding H, Xu L. Short versus standard implants for single-crown restorations in the posterior region: A systematic review and meta-analysis. Journal of Prosthetic Dentistry 128: 541.
2. Hassan H, Barakat S, Sarhan Q (2021) Survey on serverless computing. Journal of Cloud Computing 39.
3. Denninnart C, Amini Salehi M (2022) SMSE: A Serverless Platform for Multimedia Cloud Systems. arXiv.org [Online]. Available: https://export.arxiv.org/pdf/2201.01940v2.pdf.
4. Sentinel O (2021) Function as a Service (FaaS) Explained. Sentinelone. Available: https://www.sentinelone.com/blog/function-as-a-service-faas/
5. Song W (2022) Benchmark (Android apps) used for evaluating PermDroid.
6. Yuan T, Ma JQ, Zhong YP, Zhang SY (2020) Key Management Scheme Using Time-Based Deployment for Wireless Sensor Networks. Journal of Software 21 516-527.
7. Jiang L, Pei Y, Zhao J (2020) Overview Of Serverless Architecture Research. 1453: 012119.
8. Bezverbnyi I, Shyshkina M (2020) The Use of Serverless Technologies to Support Data Processing within the Open Learning and Research Systems. Symposium on Advances in Educational Technology 489-494.
9. Annugula K (2022) Software Configuration Management in Software Product Lines: Results of a Systematic Mapping Study. IEEE Latin America Transactions 20: 718-730.