Software Architecture Design of a Serverless System

Muhammad Hamza
Software Engineering LUT, University, Lappeenranta, Finland
muhammad.hamza@lut.fi

ABSTRACT

Context: Serverless computing allows developers to create and deploy applications without the need to manage any underlying infrastructure, making it a more efficient and effective way to bring products to market. Serverless technology is gaining widespread adoption among many companies, becoming increasingly popular. However, the adoption of serverless technology brings with it several new challenges. Objective: To this end, we plan to gain a deep understanding of challenges and strategies, architectural issues and their causes, architectural patterns, antipatterns, migration towards serverless architecture, and state-of-the-art practices for vendor lock-in problems. Methodology: The research objective will be met through the use of an industrial empirical approach, including interviews, a case study, and a questionnaire survey. Possible outcomes: The expected outcomes would be (i) a multivocal literature review on design areas of serverless architecture (ii) an evidence-based framework for synthesizing serverless architectural challenges/solutions (iii) a decision-making process for migrating to serverless architecture (iv) a decision-making framework for selecting vendor platform.

CCS CONCEPTS

• **Software and its engineering**; • **Human-centered computing**;; • **Computing methodologies**; • **Artificial intelligence**;; • **Philosophical/theoretical foundations of artificial intelligence**;; • **Social and professional topics**; • **Empirical studies**;

KEYWORDS

Serverless Architecture, Decision Model, Empirical Investigation

ACM Reference Format:
Muhammad Hamza. 2023. Software Architecture Design of a Serverless System. In *Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE '23), June 14–16, 2023, Oulu, Finland.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3593434.3593471

1 INTRODUCTION

Serverless computing is a revolutionary approach to building and deploying applications that abstract the need for companies to manage any underlying infrastructure. Serverless allows developers to concentrate on the business logic of their applications, while the cloud service provider handles all the overhead of monitoring, provisioning, scaling, and managing the infrastructure. This pay-as-you-go model allows companies to pay for the computational time they use [1]. Serverless computing platforms such as AWS Lambda, Azure Functions, and Google Cloud Functions have been introduced to the market. According to the report [2], the market size of serverless computing will grow to $22 billion by 2025.

Function-as-a-Service (FaaS), also known as "serverless functions," is the most widely adopted form of serverless computing. With FaaS, developers simply need to upload the source code of short-running functions and specify triggers to execute them. The FaaS provider will then execute and charge for these functions, scaling them up or down as needed. Despite being a relatively new concept, serverless computing has been successfully implemented in a variety of real-world applications, such as online collaboration tools and the Internet of Things (IoT).

However, the emergence of this new paradigm and the way it is developed differ significantly from traditional cloud computing. Thus, it may face numerous technical and operational challenges including cold start performance, the use of programming frameworks, and testing and debugging [3],[4]. There have been several studies on different aspects of serverless computing, including serverless architecture design [5], development features and limitations [6], technology aspects [7], performance properties of serverless platforms [8], and characteristics of serverless applications [9]. These studies have used various methods such as surveys, literature reviews, and evaluation measurements. For example, Li et al. [5] provided a detailed overview of serverless architecture, including its concepts, pros and cons, and implications for architecture. Similarly, Yussupov et al. [7] conducted a comprehensive technology review to compare and analyze the ten most prominent serverless platforms in terms of development, event sources, observability, and access management. Leitner et al. [10] conducted a mixed-method study on serverless patterns by examining both peer-reviewed and gray literature. They aimed to identify the types of applications that benefit most from serverless technology, and the patterns used to implement them. They identified 5 patterns for implementing serverless-based services and discussed the potential drawbacks in terms of development overhead, configuration, and performance for each pattern.

However, the current literature does not examine empirically the process of migration of any architecture to serverless computing, architectural issues, and their causes that developers face while developing serverless applications, evaluation of the serverless architecture characteristics, patterns and antipatterns, and tactics that have been proposed to improve the quality attributes of the serverless system. Furthermore, there is a lack of a decision model to assess the capability of using serverless computing.

EASE '23, June 14–16, 2023, Oulu, Finland

ACM ISBN 979-8-4007-0044-6/23/06.
https://doi.org/10.1145/3593434.3593471

Table 1: Research Questions

Research Questions	Objectives
RQ1: What are the pains and gains of developing applications for serverless architecture?	This research question aims to investigate the challenges and solutions particularly related to each design area of serverless architecture. we will conduct a multivocal literature review to meet the objective.
RQ2: What architectural issues and their causes do developers face while developing serverless applications?	The objective of this research question is to explore the architectural issues and their causes that developers face while developing serverless applications. We will explore Github projects as a source for identification.
RQ3: What are the gaps between the vision and reality of a serverless system?	This RQ aims to: (1) investigate real concerns and practices in terms of the specific visions (characteristics) claimed of serverless architecture between different domains; (2) compare the results around different domains; and (3) generate an overview map about practices to guide practitioners.
RQ4: What are the architectural patterns and antipatterns for serverless systems? A state-of-the-art practice.	This research question aims to identify the architectural patterns being implemented in the industry and anti-patterns that are avoided to make the serverless application successful.
RQ5: What is the decision-making process of organizations during migration towards any architecture to serverless?	The objective of this research question is to empirically identify the decision that an organization makes before migrating to serverless architecture, at what point in the migration are decisions made and what alternatives do organization choose in each step.
RQ6: How to select the vendor platform for serverless applications.	This research question will lead to the development of a decision model that will assist organizations to select the vendor for serverless applications based on organizational requirements. The developed model will be evaluated with an industrial case study.

To the best of our knowledge, no study empirically evaluated the aforementioned limitation of research. Therefore, we have decided to conduct comprehensive research in these areas as a Ph.D. dissertation.

2 RESEARCH DESIGN

The objective of this research proposal is to empirically investigate different aspects of serverless architecture. To meet the objective, the main research questions are formulated as follows.

3 RESEARCH METHOD

3.1 Multivocal Literature Review

A multivocal literature review (MLR), includes sources such as industrial reports, government documents, organizational reports, and publicly available resources like blogs and online articles. This type of literature is often more prevalent in new or emerging fields, where there may be less peer-reviewed research available. Unlike a traditional systematic literature review (SLR), which only considers peer-reviewed research, an MLR considers a wider range of literature and can be valuable in answering questions from both practical and industrial viewpoints. To conduct the multivocal literature review, we will follow MLR protocols proposed by Garousi et al. [11]. We will conduct MLR to identify the challenges and their solutions in each design areas of serverless architecture.

3.2 Survey

A survey is a method of collecting data from a sample of individuals through the use of standardized questions. This is the second research method that would be used to explore and validate the challenge and solutions from the industrial practitioners who are involved in developing serverless applications. Given the potential presence of multiple variables and the need to test several hypotheses in order to identify challenges and solutions related to serverless architecture, we have determined that the use of a cross-sectional survey would be appropriate in this situation. This type of survey allows us to gather data from a sample of individuals at a specific point in time, which will enable us to examine relationships between variables and test hypotheses.

3.3 Qualitative Analysis

Qualitative research methods include techniques such as interviews, focus groups, and observations. It is often used to explore and understand people's experiences, perspectives, and behaviors in a particular context. We will use qualitative methods to study the strategies that serverless application developers employ to cope with vendor lock problems. Furthermore, we will employ qualitative methods for addressing research question (RQ3 to RQ6).

3.4 Case Study

There is currently a lack of empirical research on serverless architecture, and we believe that exploratory case studies can help to deepen our understanding of this topic and provide insights into its use in the industry. Case studies allow us to consider multiple sources of evidence in relation to a small number of specific instances. In addition, the context and actors within an organization are important factors in our research questions, making the use of "multiple embedded case studies" an appropriate method for collecting data in this study [12].

3.5 Data analysis methods

We anticipate that our research will generate a large volume of both qualitative and quantitative data from the multivocal literature review, surveys, and case studies. To analyze the qualitative data, we will use the "constant comparison" method, which involves continuously comparing data within and across cases. For the quantitative data, we will use statistical techniques such as measures of central tendency, and measures of variability, such as variance and standard deviation. To ensure the validity of our findings, we also plan to send the results of the study to willing participants for review.

4 EXPECTED OUTCOMES

The expected outcomes of this research project are manifold: (i) A multivocal literature review on the design areas of serverless architecture (ii) an evidence-based framework for synthesizing serverless architectural challenges/solutions (iii) a decision-making process for migrating to serverless architecture (iv) a decision-making framework for selecting vendor platform.

REFERENCES

[1] R. A. P. Rajan, "Serverless architecture-a revolution in cloud computing," in 2018 Tenth International Conference on Advanced Computing (ICoAC), IEEE, 2018, pp. 88–93.

[2] "A research and markets report," May 2022. [Online]. Available: https://www.researchandmarkets.com/reports/4828585/serverless-architecture-market-by-deployment

[3] E. Jonas *et al.*, "Cloud programming simplified: A berkeley view on serverless computing," arXiv preprint arXiv:1902.03383, 2019.

[4] L. Wang, M. Li, Y. Zhang, T. Ristenpart, and M. Swift, "Peeking behind the curtains of serverless platforms," in 2018 USENIX Annual Technical Conference (USENIX ATC 18), 2018, pp. 133–146.

[5] Z. Li, L. Guo, J. Cheng, Q. Chen, B. He, and M. Guo, "The serverless computing survey: A technical primer for design architecture," ACM Computing Surveys (CSUR), vol. 54, no. 10s, pp. 1–34, 2022.

[6] T. Back and V. Andrikopoulos, "Using a microbenchmark to compare function as a service solutions," in European Conference on Service-Oriented and Cloud Computing, Springer, 2018, pp. 146–160.

[7] V. Yussupov, J. Soldani, U. Breitenbücher, A. Brogi, and F. Leymann, "FaaSten your decisions: A classification framework and technology review of function-as-a-Service platforms," Journal of Systems and Software, vol. 175, p. 110906, 2021.

[8] W. Lloyd, S. Ramesh, S. Chinthalapati, L. Ly, and S. Pallickara, "Serverless computing: An investigation of factors influencing microservice performance," in 2018 IEEE international conference on cloud engineering (IC2E), IEEE, 2018, pp. 159–169.

[9] S. Eismann *et al.*, "The state of serverless applications: Collection, characterization, and community consensus," IEEE Transactions on Software Engineering, vol. 48, no. 10, pp. 4152–4166, 2021.

[10] P. Leitner, E. Wittern, J. Spillner, and W. Hummer, "A mixed-method empirical study of Function-as-a-Service software development in industrial practice," Journal of Systems and Software, vol. 149, pp. 340–359, 2019.

[11] V. Garousi, M. Felderer, and M. V. Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," Information and Software Technology, vol. 106, pp. 101–121, 2019.

[12] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, "Selecting empirical methods for software engineering research," in Guide to advanced empirical software engineering, Springer, 2008, pp. 285–311.