

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/328216639>

# Challenges of Microservices Architecture: A Survey on the State of the Practice

Article · May 2018

CITATIONS

4

READS

1,153

2 authors:



Javad Ghofrani

Hochschule für Technik und Wirtschaft Dresden

17 PUBLICATIONS 26 CITATIONS

SEE PROFILE



Daniel Lübke

Leibniz Universität Hannover

62 PUBLICATIONS 573 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Executable Business Process Testing [View project](#)



clone detection [View project](#)

# Challenges of Microservices Architecture: A Survey on the State of the Practice

Javad Ghofrani and Daniel Lübke

Leibniz Universität Hannover, Hannover, Germany,  
{javad.ghofrani,daniel-luebke}@inf.uni-hannover.de

**Abstract.** Microservices have been one of the fastest-rising trends in the development of enterprise applications and enterprise application landscapes. Even though various mapping studies investigated the open challenges around microservices from literature, it is difficult to have a clear view of existing challenges in designing, developing, and maintaining systems based on microservices architecture as it is perceived by practitioners. In this paper, we present the results of an empirical survey to assess the current state of practice and collect challenges in microservices architecture. Therefore, we synthesize the 25 collected results and produce a clear overview for answering our research questions. The result of our study can be a basis for planning future research and applications of microservices architecture.

**Keywords:** Empirical software engineering, State of Practice, Microservices, Software Architecture

## 1 Introduction

Microservices architecture (MSA) is based on a share-nothing philosophy and relies on a long evolving experience in software engineering and system design. This architectural style structures a system as a set of loosely-coupled small services which are isolated in small coherent and autonomous units [9].

Many organizations, such as Amazon, Netflix, and the Guardian, utilize MSA to develop their continuous delivery of large and complex applications while providing flexibility and diversity of technology stack [2]. In addition to structuring the development of the systems, design principles of MSA are used to migrate systems with traditional architectural style into MSA. Since MSA is still young, there are many open issues and optimization possibilities in this field. Even though many research papers (e.g., Vural et al. [8]) tried to identify the issues and proposed solutions from literature around MSA, many aspects of the practical challenges in MSA are still unexplored. This makes it difficult for researchers to have a realistic overview of the real challenges encountered in practice related to MSA and their potential for academic and industrial adoption [2]. The goal of this paper is to characterize the current state of the practice about MSA and provide a complementary overview on the existing scientific research. In order to achieve this goal, we conducted an online survey. Specifically, we

selected three research questions related to the existing challenges and solutions from MSA in the practice. **RQ1:** *What are the main challenges/concerns in the design and the development process of microservices?* **RQ2:** *What are the main reasons leveraging and preventing the usage of systematic approaches in microservices architectures?* **RQ3:** *Are there any suggestions or solutions from the experts to improve aspects of the microservices architecture?*

The main contributions of this paper are: (i) providing an up-to-date map of the state of the practice in MSA and its complexities for future research; (ii) an evaluation of the potential research field on MSA. The audience of this paper are both researchers interested to supplement and update their overview of most recent challenges in this field for their future contributions as well as practitioners interested to get an overview of existing challenges and solutions, thereby making better decisions for their organization.

The rest of the paper is organized as follows. Section 2 gives an overview on the existing reviews from literature, whereas the design of our study is presented in Section 3. We elaborate the results of our study in Section 4 by putting them in a broader perspective for answering the proposed research questions. With Section 5 we conclude the paper and discuss the future work. The full survey data including the questionnaire and preprocessed results is shared and available online [5].

## 2 Related Work

As described in the previous section, the existing academic studies reviewed the literature to provide an overview of open challenges in MSA. There is no academic survey on the state of the practice in MSA as of now. As a basis for our study, we consider the following systematic literature reviews and mapping studies conducted about state of the art in MSA.

Pahl and Jamshidi [7], Alshuqayran et al. [1], and Di Francesco et al. [2] conducted systematic mapping studies about MSA. Pahl and Jamshidi [7] investigated 21 publications until 2015 and reported the existing research trends and directions in this field with regards to applications in the cloud. The informal survey of Dragoni et al. [3] around MSA opens an academic viewpoint to the open problems for novices. The main difference between our study and [1–3, 7] is the research approach, which is used to provide an overview of open issues and challenges in MSA. The previous approaches prefer the academical aspect of existing literature while we explore the practical side of the domain. The provided results by Pahl and Jamshidi [7] are subject to the passage of time and are perhaps even outdated because of the fast progress of MSA’s underlying technologies.

## 3 Study Design

Since surveys provide a better overview of usage of technology in the software industry [4], we performed an online survey to get the practical picture of MSA.

We designed our survey in a brain storming session with a group of five researchers and created a mind-map. We created a list of questions and possible answers based on our mind-map. An iterative review process is performed to assure that the survey questions are comprehensive and relevant to the domain. Finally, we performed pilot tests with two other volunteers.

We created our online survey based on the guidelines described by Jacob et al. [6]. Our survey is mainly based on using multiple choice and numerical questions to provide an “other” option to consider missing options. The answers of “other” options are used either to create new categories or considered as belonging to one of the existing categories by analyzing the survey results. Self-containment of the survey questions are also adhered by providing required information about the context or expressions of a question. In order to avoid long questionnaire for the intended audience, we split the questionnaire into five parts: Profiling, Domain, Modeling, Non-functional, and a Final part. The participants were informed about the results of the survey after leaving their email addresses in the final step of the survey. Our survey was available on the LimeSurvey Server, hosted by Software Engineering Group of Leibniz Universität Hannover, between November 17, 2017 and January 5, 2018.

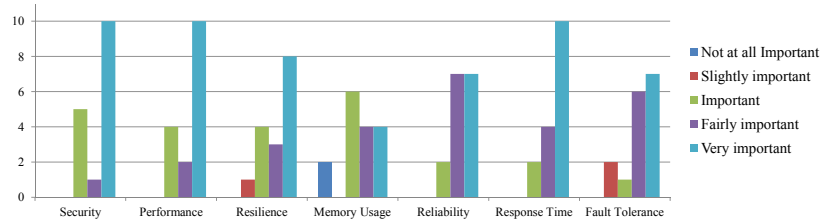
First, we reached the potential participants via our personal contacts and asked them to take part and also forwarded our invitations to potential participants that they may know. In addition, we posted the survey invitation to several online communities, XING, meet-up groups, and advertised the survey via mailing lists of the NGINX, Inc. In order to control who was answering our survey, we started our survey with a filter question about taking part in a project with relevance to MSA.

## 4 Results

In total, 40 experts participated in our survey and 25 of them has answered *yes* to the filter question about being involved in any project or program that are related to the microservices architecture. Since our research is related to the practical part of the MSA, we consider only the answers of the participants who answered *yes* to the filter question and answered at least one of the other questions from our survey. The Majority of the respondents, 19 (76%), are practitioners from industry whereas only 2 of them (0.08%) are from academia. Furthermore, 4 participants are active in both academic and industrial contexts. By asking about the role of the participants in their current activities, 17 (68%) stated that they are developers, 11 (44%) system architects, and 3 (12%) of them are team leaders. All of the team leaders and 8 of the system architects are developers simultaneously. Relative to experience with MSA, 11 (44%), have between 1 and 2 years, and 8 (32%) have between 2 and 5 years of experience. 21 participants stated that they use agile methods in their organizations.

#### 4.1 RQ1: What are the main challenges/concerns in the design and the development process of microservices?

To answer RQ1, we asked our participants about their main challenges in the development process of microservices. According to the 8 comments and free-text answers of the survey to this question, the distributed nature of the MSA is one of the main challenges in the development and debugging of the systems based on this architectural style. “Too many repositories to maintain”, “Hard to find issues in a distributed system. Rarely any benefit”, “networking between dockers”, “sharing base data among different services considering performance limitations”, and “debugging a microservice that relies on other services can be tricky” are some of these comments. Next common challenge is the skill and knowledge. It seems that there are difficulties between customers and development teams in “understanding MS of the Top Management/Director levels”, “getting the right developers/engineers”, and “Changing the people’s minds that are used to traditional monoliths”. Finally, “correct separation of domains” and “finding the appropriate service cuts” are the third majority of the concerns stated by our participants. Nevertheless, they have also recommended to use MSA instead of other architectural approaches in case of “Distributed teams that can have logical separations”.



**Fig. 1.** Priority of MSA features that should be optimized, according to the number of votes given by practitioners

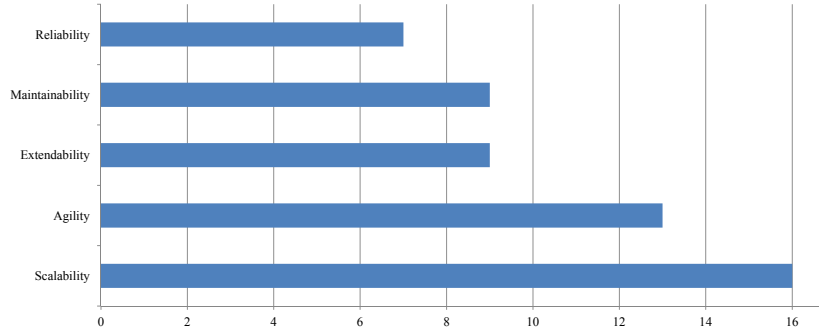
In order to get an overview of the main concerns of our participants regarding nonfunctional features of their MSA, we asked “How important is it for you to optimize the following features of your microservice architecture?”

- Security
- Performance
- Resilience
- Memory Usage
- Reliability
- Response time
- Fault tolerance <sup>1</sup>

<sup>1</sup> We use 5 levels of importance: Not at all important, Slightly important, Important, Fairly Important, and Very important

Fig. 1 depicts the frequency and the importance level of each the features. One can see that Security, Performance, and Response Time are mentioned as very important to optimize in MSA. The optimization of Resilience, Reliability and Fault Tolerance are considered to be the next priorities while the Memory Usage is only important.

Furthermore, we asked our participants “What were your goals when deciding on a microservice architecture?”. Fig. 2 shows that Scalability (16/25) and Agility (13/25) are commonly desired compared to Extendability (9/25), Maintainability (9/25), and Reliability (7/25).

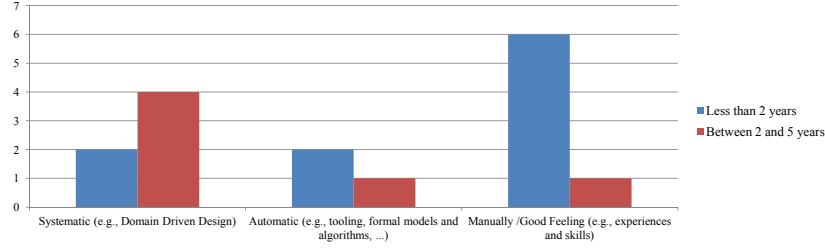


**Fig. 2.** Participants’ primary intention for using a microservice architecture

#### 4.2 RQ2: What are the main reasons leveraging and preventing the usage of systematic approaches in microservices architectures?

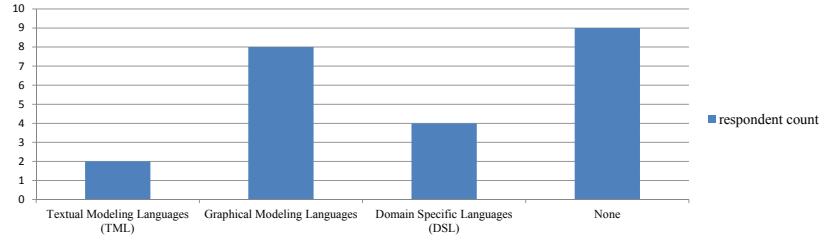
We asked the participants “How do you derive service boundaries?”. On the one hand, a big amount of the participants (7/25) use Manually / Good Feeling methods based on their experiences and skills. Looking at the skills of these participants reveals that most of them have less than 2 years of experience with MSA. On the other hand, 6/25 of participants inform us that they use systematic approaches, especially Domain Driven Design, for deriving the service boundaries. It shows a significant correlation (85%) with their experience on MSA, between 2 to 5 years. Automatic methods, such as tooling, formal methods, and algorithms for deriving service boundaries were used by 3 participants. Two of them have less than 2 years of experience in MSA and only one has between 2 and 5 years of experience. These answers are depicted in Fig. 3.

The next question was: “Which notation(s) do you use to describe your architecture?”. As shown in Fig. 4, 8/25 of the participants stated that they use Graphical Modeling Languages (GML) because of agility, simplicity, and easily available GML tools. 2/25 use Textual Modeling Languages (TML) and



**Fig. 3.** Popularity of each method among the participants with less than 2 years (blue bars) and between 2 to 5 years (red bars) of MSA experience

4/25 of them use Domain Specific Languages (DSL). Nevertheless, 9/25 of participants stated that they do not use any notation in this context. None of the participants named any tool, technique, and framework for modeling their MSA which indicates the limited awareness on existing tools for MSA among the practitioners.



**Fig. 4.** Notations that are used by practitioners to describe their MSA, according to 25 collected responses

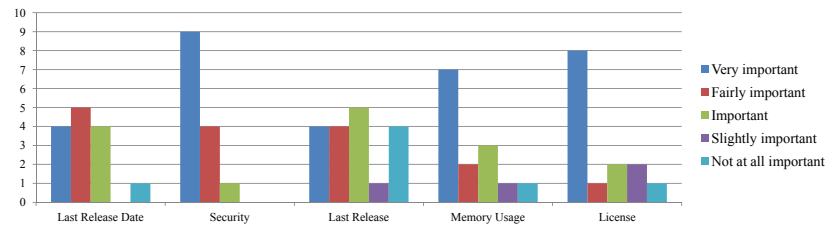
#### 4.3 RQ3: Are there any suggestions or solutions from the experts to improve aspects of the microservices architecture?

Finding a trade-off between reusing and developing the architectural artifacts is a well-known issue in any software architecture. We asked the participants about their opinion on usage of artifacts from third-parties within a MSA as the following question: “If you use any artifacts from third-parties in your microservice architecture, do you consider the following features? How important are they for you?”.

- Last Release Date
- Security

- Last Release
- Memory Usage
- License <sup>2</sup>

As Fig. 5 illustrates, Security, License, and Memory usage are the most respected concerns according to the given priorities. Furthermore, one respondent mentioned that “he/she often finds out about memory footprint or computing time only after he/she has integrated and tested” about third-party frameworks or libraries. There is also an interesting suggestion from a respondent to start renaming microservice nomenclature in order to facilitate the utilization of Domain Driven Design tools within the MSA context.



**Fig. 5.** The importance of various aspects of artifacts from third-parties according to the collected responses

## 5 Conclusions and Future Work

In this paper we reported the results of our online survey that we conducted among experts of Microservices architecture (MSA). We have investigated survey responses to answer our three research questions about challenges in MSA. Since the majority of our experts are from industry, our research reflects the practical issues in MSA. Our work can be used as a complementary work to the existing literature reviews to guide researchers to the open issues and problems in MSA and offer an overview from practical point of view.

Among the collected responses, the lack of notations, methods, and frameworks to architect MSA can be considered as important points. The lack of tool or framework support for selecting third-party artifacts according to their features, e.g. security, last release, as well as the shortage of knowledge of the practitioners about systematic methods are the top main gaps, which can be addressed in the future work. According to the results of our survey, optimization in Security, Response Time, and Performance have higher priorities than Resilience, Reliability, Fault Tolerance, and Memory Usage.

<sup>2</sup> We use 5 levels of importance: Not at all important, Slightly important, Important, Fairly Important, and Very important



Future work includes (i) conducting a survey each year to monitor the change over time, (ii) performing a new survey with a larger scope to collect more details about further aspects of MSA in the practice and (iii) proposing solutions for the identified gaps.

## References

- [1] Alshuqayran, N., Ali, N., Evans, R.: A systematic mapping study in microservice architecture. In: Service-Oriented Computing and Applications (SOCA), 2016 IEEE 9th International Conference on. pp. 44–51. IEEE (2016)
- [2] Di Francesco, P., Malavolta, I., Lago, P.: Research on architecting microservices: Trends, focus, and potential for industrial adoption. In: Software Architecture (ICSA), 2017 IEEE International Conference on. pp. 21–30. IEEE (2017)
- [3] Dragoni, N., Giallorenzo, S., Lafuente, A.L., Mazzara, M., Montesi, F., Mustafin, R., Safina, L.: Microservices: yesterday, today, and tomorrow. In: Present and Ulterior Software Engineering, pp. 195–216. Springer (2017)
- [4] Fink, A.: The survey handbook, vol. 1. Sage (2003)
- [5] Ghofrani, J., Lübke, D.: Online material for survey on challenges of microservices architecture (2018), <https://doi.org/10.6084/m9.figshare.5852598>
- [6] Jacob, R., Heinz, A., Décieux, J.P.: Umfrage: Einführung in die Methoden der Umfrageforschung. Walter de Gruyter (2013)
- [7] Pahl, C., Jamshidi, P.: Microservices: A systematic mapping study. In: CLOSER (1). pp. 137–146 (2016)
- [8] Vural, H., Koyuncu, M., Guney, S.: A Systematic Literature Review on Microservices, pp. 203–217. Springer International Publishing, Cham (2017)
- [9] Wolff, E.: Microservices: Flexible Software Architecture. Addison-Wesley Professional (2016)