Selection Methods for Criticality in Microservices Architectures: A Systematic Mapping

Eduardo F. Mioto de Oliveira dos Santos[1], Claudia Maria Lima Werner[2], Marcelo de França Costa[3]

Universidade Federal do Rio de Janeiro, Rio de Janeiro RJ 08544, Brazil
{mioto[1], werner[2]}@cos.ufrj.br, aluno@franca.pro.br[3]

Abstract. The selection method for criticality associated with Microservices Architecture (MSA) has great importance associated with system reliability. Since MSA is a new research area, the need for a systematic literature review is crucial in order to elucidate the possible ways to identify consolidated criticality methods. The proposal of this systematic mapping is to identify the gaps and approaches used on previous similar situations on studies related to MSA, also including SOA and components, considering these being part of microservices background and evolution. Thus this study focuses on identifying methods to evaluate a decision focused on highlighting the most critical component, service or MSA. In this paper, 68 primary studies about this topic published from 2012 to 2019 were selected. Results indicate that the software engineering community has increasingly invested effort in studying criticality evaluation. Nevertheless, it is still an immature area when focused on MSA.

Keywords: Microservices, SOA, Components, Selection, Multiple Criteria, Criticality, Systematic Review

1 Introduction

According to Haselböck, Weinreich, and Buchgeher [7], introducing a Microservice Architecture (MSA) is not an effortless journey, a company undertaking such an endeavor needs to make a variety of decisions considering a wide range of development practices, technology, and infrastructure to handle the additional challenges and complexity that accompany this architectural style.

MSA arises from the broader area of Service Oriented Architecture (SOA) and focuses on specific aspects, such as componentization of small services, application of agile practices for development, deployment, and testing of services [18]. According to Montesi and Weber [16], the key difference between MSA and SOA lies in granularity. Gabbrielli et al. [17] also mention that scaling a microservice architecture does not imply duplication of all its components and developers can conveniently deploy or dispose of instances of services in accordance with their load.

MSA development is an architectural model to be followed to designing software applications as independent service groups. These applications are cohesive and small in size, and can be easily integrated with reduced creation, validation and deployment time. Microservices can be updated and sized individually, leading to greater architectural stability and greater resilience [22].

The need to know the most critical microservices comes from the complexity in decision making regarding the maintenance and evolution of architectures based on this architectural model. After a microservice oriented architecture deployment, the architecture maintenance process tends to be complex, considering the multiple services to be observed.

Francesco, Malavolta and Lago [13] mention that the need to understand the current state of the art on architecting microservices is confirmed by the fact that three different mapping studies [13, 14, 15] have been published recently and are providing important characterization to the field, although, none of them associated with criticality to MSA.

This paper contextualizes the use of decision models to support the establishment of an MSA considering the criticality evaluation methods. Decision models represent a well-known approach for exploring the design space, making decisions, documentation, and reuse in software architecture [7, 8, 9].

Since MSA is an architectural style, the objective of this study is to explore how previous research has supported microservices through architectural approaches. The aim of the study is to focus on the research questions exposed on the Section (2.1), closely link and correlate the research questions to the review study results and provide quantified evidence from the available publications.

The contribution of this paper is a comprehensive overview of existing approaches on selection methods for criticality in MSA developed over the last years and their focus in terms of the different activities. The criticality analysis is a process through which artifacts are evaluated in terms of potential risk of degradation, partial failure or total failure [20]. On software context, performing a criticality analysis helps to clarify the available methods to reduce the risk associated with each software asset.

Three main aspects distinguish this work from previous, similar work is this area. First, approaches for components and SOA based on published studies were identified, by aggregating related studies that considered the same approach or variations. Second, for each approach, the focus of the performed research was determined. Finally, the provided evidence in terms of associated studies not related to general approach was identified.

The remainder of this paper is structured as follows. Section 2 presents the research method, elucidating the research questions, the search process, and the inclusion and exclusion criteria, also exposing the search string, as well as detailing the quality assessment and data collection. In Section 3, the results and their implications are detailed, also considering the data extraction and research questions answering, as well as the threats to the validity and study limitations. In Section 4 the related works are listed. The paper is concluded in Section 5, which presents the conclusion and future researches.

2 Research Method

Based on the methodology of systematic literature reviews (SLR) [1, 2, 3, 4, 5], a scheme was developed for the review, selection and extraction of information in the following phases: (i) Planning the Review, (ii) Conducting the Review and (iii) Reporting the Review.

According to the Australian National Health and Medical Research Council [1], an SLR is a means of identifying, evaluating and interpreting all available research relevant to a particular research question, topic area, or phenomenon of interest. Individual studies contributing to an SLR are called primary studies; an SLR is a form of a secondary study. This paper intends to identify gaps in the current state of the art in order to suggest areas for further investigation.

Systematic literature reviews (SLRs) are considered comprehensive and rigorous reviews of specific research questions in an area or a topic, which aim to identify the gaps in the literature and identify where new or better primary studies are needed to be put in place [18]. Systematic mapping is a secondary study method based on a structured and repeatable process or protocol that explores studies and provides a result in the form of an overview of a particular subject [23].

2.1 Research Questions

The overall goal of this systematic mapping is to identify existing MSA, SOA and components approaches and to determine their support for architecture decisions associated with criticality. Accordingly, the following three research questions (RQs) were formulated:

RQ1 - What are the established selection models for MSA, SOA, and components? In RQ1, the approaches and families of approaches to MSA, SOA, and components associated with selection models will be identified by combining related studies obtained from the literature.

RQ2 - What are the multiple criteria evaluation approaches for MSA, SOA, and components? The information from RQ2 intends to determine which microservices, SOA and components activities related to multiple criteria evaluation are well supported by current researches on literature. Therefore, gaps in current approaches and needs for future research with respect to support for MSA, SOA and components activities can be identified.

RQ3 - What are the established criticality methods in MSA, SOA, and components? With RQ3, it is desired to gain better insights into how well existing approaches have been validated both with respect to academic and industrial studies. The evidence provided for existing approaches is rated with respect to how they support the MSA, SOA and components activities related to criticality.

2.2 Search Process

Initially, a set of research questions were drafted for investigation during the study. The motivation behind each research question was reviewed and refined. Subsequently, selected papers were assessed against quality criteria and a classification scheme was iteratively developed following a synthesis method.

2.3 Inclusion and exclusion criteria

It was defined a number of criteria on the basis of which papers were included in the review or excluded from the review. The defined inclusion (I1-I3) and exclusion (E1-E7) criteria are exposed in Table (1).

Table 1. Inclusion and Exclusion criteria

Criteria Type	***Criteria Description***
Inclusion (I1)	The paper is related to microservices selection, decision making or criticality
Inclusion (I2)	The paper is related to SOA services selection, decision making or criticality
Inclusion (I3)	The paper is related to components selection, decision making or criticality
Exclusion (E1)	The paper is not related to MSA, SOA or components
Exclusion (E2)	The paper is not related to selection, decision support or criticality
Exclusion (E3)	The paper is based on Gray Literature
Exclusion (E4)	The paper was not found.
Exclusion (E5)	The paper was not found without cost.
Exclusion (E6)	The paper is not written in English.
Exclusion (E7)	The paper does not represent a primary study

During the study selection phase, some filtering information was also considered to be able to get just the most updated information regarding the research area. These filters were:

- Qualitative studies will be considered.
- Language: English only
- Years: Between 2012 and 2019
- Document Types: All types of articles except gray literature.
- Outcome Measure: Number of identified methods
- Keywords: microservices, SOA, components, architecture, criticality, AHP, MDS
- Search Engines: Scopus, IEEE, Web of Science, Springer and Science Direct
- Experimental Design: there will not be applied to statistical methods

During the evaluation of the qualitative studies, it was also considered a duplicity filter. Thus after adapting the search criteria to the filters mentioned above, having run the inclusion and exclusion criteria, all studies returned by the search that are not duplicitous were accepted. At the end of the bibliographical survey, it is expected to identify the state of the art of criticality treatment models in microservices oriented architectures.

2.4 Conducting the Systematic Mapping

In this phase, the process specified in the protocol previously described was carried out. The main steps were: (i) Identification of Research, (ii) Selection of Primary Studies, (iii) Data Extraction, and (iv) Data Analysis based on the method used by Khan et al [4]. On digital libraries, the search string shall be adapted to each engine in order to obtain suitable results.

Table 2. Search String

Search String: *"prioritize and microservices"* **OR** *"ahp and microservices"* **OR** *"mds and microservices"* **OR** *"multidimensional and microservices"* **OR** *"multicriteria and microservices"* **OR** *"decision and microservices"* **OR** *"selection and microservices"* **OR** *"ahp and architectures"* **OR** *"mds and architectures"* **OR** *"multidimensional and architectures"* **OR** *"multicriteria and architectures"* **OR** *"decision and architectures"* **OR** *"selection and architectures"* **OR** *"criticity and microservices"* **OR** *"criticality and microservices"*

Selection of Primary Studies: In the first filter, the title, abstract, keywords of the papers were read. Based on this information, the inclusion and exclusion criteria were applied. After this step, 16 papers were classified as duplicated, 451 were rejected and 140 were accepted.

During the second filter, named as the extraction phase and detailed in Section (3), 140 accepted studies were analyzed considering the entire paper content. At the end, a total of 68 studies were accepted, a total of 523 studies were rejected, and 16 studies were considered duplicated.

2.5 Quality Assessment

The authors Kitchenham et al. [6] propose four quality questions for evaluating SLR. These are presented in the following, along with an evaluation of this study against these questions:

QA1: Are the reviews inclusion and exclusion criteria described and appropriate? It was explicitly defined and discussed the inclusion and exclusion criteria used in Section (2.3), so this quality criterion is met.

QA2: Is the literature search likely to have covered all the relevant studies? According to Kitchenham et al. [6], this criterion is met if four or more digital libraries have been searched and additional search strategies have been included. This quality criterion is

met, as it was performed an automatic search in five scientific databases (Scopus, IEEE, Web of Science, Springer and Science Direct).
QA3: Did the reviewers assess the quality/validity of the included studies? It was explicitly assessed the quality of each primary study according to the defined quality criteria. As part of the data synthesis, it was explicitly validated the level of evidence presented for each approach with respect to the MSA, SOA and components activities each supported.

QA4: Were the basic data/studies adequately described? This quality criterion was considered met since a detailed data collection form was used for each study. The results of merging papers to approaches, their focus, and support in terms of the different MSA, SOA, and components activities were also grouped for summarizing.

2.6 Data Collection

To answer the proposed research questions, each publication from the final set of papers was read in detail, extracting specific data about the presented approach or concept. In addition to general information about the selected paper, data regarding the underlying microservices, SOA and components model in the decision-making process were extracted.

To systematically search for scientific sources, population, intervention, comparison, and outcome (PICO) criteria to define the terms for the database search were used. The population in this paper is represented by components, SOA and MSA papers because it is only interested in criticality approaches applicable to these areas. The intervention is architectural knowledge and any synonyms thereof. Comparison is not applicable in this case, as this paper did not take a comparative approach. The search source was selected based on the capability to research reproducibility and indexing of the main periodicals and publications of the area.

3 Results and Discussion

In this section, the answers to the research questions are discussed, considering the broad-spectrum of approaches.

3.1 Search Results

This section presents the results of search and selection. Table (3) exposes the paper selection status separated by the search engine. After the extraction phase mentioned in Section (2.4), the studies were classified by publication year and publishing vehicle. In total, 68 selected studies were included, 8 being published in journals and 60 published in conference proceedings.

Table 3. Papers selection status per search engine

Engine	Articles			
	Total	***Accepted***	***Rejected***	***Duplicated***
Scopus	101	19	81	1
IEEE	164	24	126	14
Web of Science	75	7	68	0
Springer	258	13	245	0
Science Direct	9	5	3	1

It was obtained a total of 607 articles after search phase and accepted a total of 68 articles

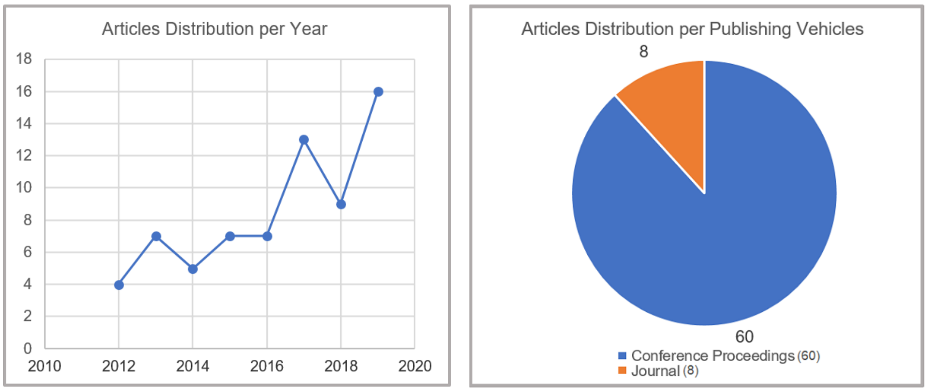

Figure 1. Papers by Year and Type of Publishing Vehicles

Figure (1) shows the distribution of studies included in this paper. The distribution of studies over the years, from 2012 to 2019. This figure shows a trend of increasing publications on MSA, SOA and components approach between 2017 and 2019, in which publications on MSA, SOA, and components approaches are published with a slight variation in 2018, but with an increasing tendency.

Other studies on similar topics also show that MSA, SOA and components and related topics such as design decisions and documentation are still an active research discipline in general. All of them show a similar increase after 2015 with slight increases and decreases in the following years. A full list containing the selected articles is available on-line (http://www.eduardomioto.com/review/cibse2020.htm).

3.2 Selection Methods for MSA, SOA, and Components (RQ1)

As regards to selection methods, 18 papers did not propose any selection method. Considering the approaches to MSA, SOA, and components that have been used by the research community over the years between 2012 and 2019, 15 different selection methods to MSA, SOA and components were identified as shown on Figure (2).

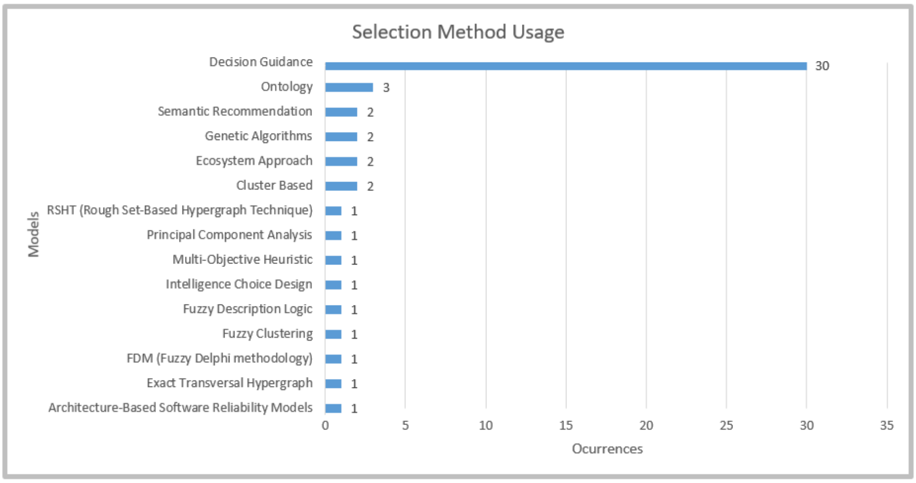

Figure 2. Number of papers per selection method

Based on these data, it was concluded that "Decision Guidance" is the most used method for selection, being used by 30 papers (44.11%) from a total of 68. The second most used selection method was "Ontology" with 3 papers, followed by "Semantic Recommendation", "Genetic Algorithms", "Ecosystem Approach" and "Cluster-Based" each of them used by 2 papers (2.94%) from the total of 68 (4.41%). All other selection models used were found in one study each (1.47% each one).

3.3 Multiple Criteria Evaluation Model (RQ2)

As regards to criticality based evaluation models, 29 papers (42.64%) did not propose any multiple criteria evaluation model. Considering the approaches to MSA, SOA, and components that have been used between 2012 and 2019, 16 different multiple criteria approaches to MSA, SOA and components were identified as shown on Figure (3).

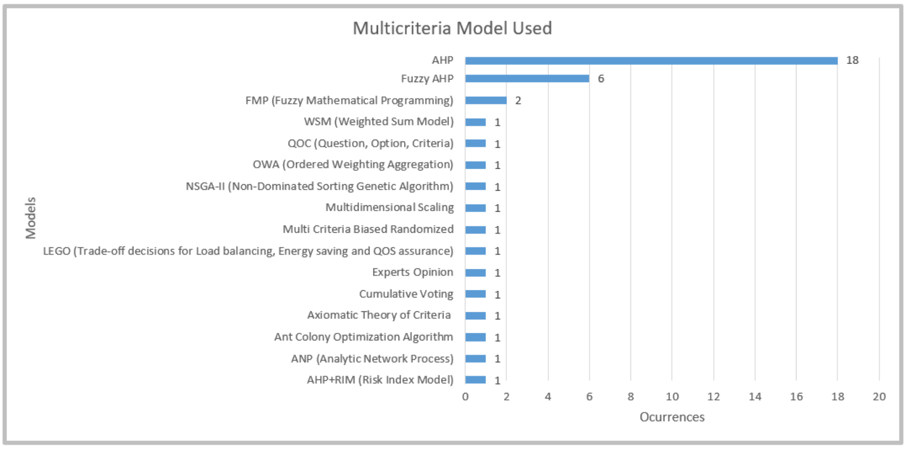

Figure 3. Number of papers per multiple criteria method

Analytic Hierarchy Process (AHP) model was used in 18 studies from a total of 68 studies (26.47%), Fuzzy AHP model was used in 6 studies (8.82%), followed by Fuzzy Mathematical Programming used in 2 studies (2.94%). All other multiple criteria models used were found in one study each (1.47% each one).

It was possible to conclude that there is a common acceptance regarding the usage of AHP as a multiple criteria approach, also considering its adaptation variations to the user requirements, as observed with Fuzzy AHP.

3.4 Criticality Based Evaluation Model (RQ3)

As regards to criticality based evaluation models, 63 papers (92.64%) did not propose any criticality evaluation model from a total of 68 selected papers. Considering the approaches to MSA, SOA and components that have been developed by the research community over the last years between 2012 and 2019, 5 different criticality based evaluation models to MSA, SOA and components were identified.

As regards to criticality based evaluation models, all occurrences were found at the same proportion, as shown in Figure (4). The model “Quality Attributes on Architecture Trade-Off Centered Analysis” was found in 1 study (1.47%) from a total of 68 studies; as well as “Dynamic Weights”, “Architecture-based Software Reliability Models”, “Components Reliability based on Defect” and “Failure Mode Effect and Analysis.

This variety of models highlights that there is not a consensus regarding the most relevant criticality evaluation model and most of them (4 out of 5) focused on components and just the minority part (1 out of 5) focused on MSA.

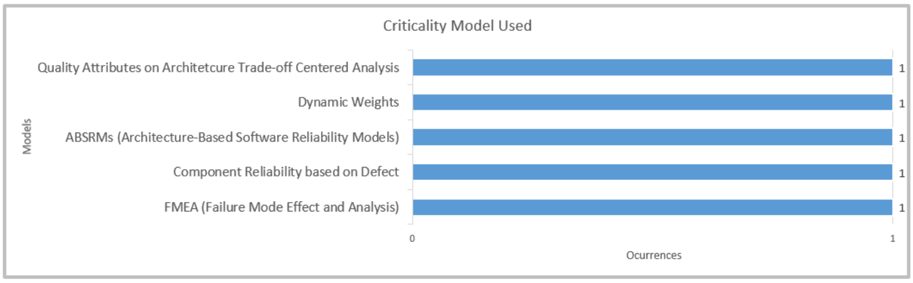

Figure 4. Number of papers per criticality method

Related to the single paper that treats the criticality focused on microservices [21], a criticality evaluation approach called ACISM was used to analyze the quality of services. The ACISM approach uses the dynamically acquired weight information to obtain the result and uses the result to select instances of microservices. According to the authors, the proposed method can weaken the decisiveness of user's subjective selection by using the dynamical weight information. The method balances the influence of subjective and objective factors and can improve the accuracy of microservice instances selection and shorten the response time.

Based on the results obtained, it was possible to conclude that there is no standardized nomenclature and semantic of concepts, which may hinder the understanding and comparison. This may be a sign of a lack of maturity in the microservices criticality research area.

3.5 Threats to Validity

In order to analyze threats to validity related to this systematic mapping, relevant concerns are described separately as follows:

Discovering the primary studies: considering the focus on the discovery of the primary study considering multiple search criteria, the search string was developed associating to synonyms and was reviewed by the second and third authors. In addition, they contain synonyms and both singular and plural forms of the main terms. However, the number of digital libraries would not consider all the content available. Finally, only papers written in English were read. Therefore, this paper may have ignored some potentially relevant primary studies.

Increasing transparency and repeatability: are improved by documenting and disseminating the research questions, search strings, inclusion and exclusion criteria, digital libraries used, and the process was undertaken. However, full transparency is not possible as some information cannot be presented in this paper, motivated by the space needed.

Conducting the search: The digital databases do not have compatible search rules, so different search string were used, but considering some keywords. It was considered to reduce any possible bias in the manner of conducting the review, avoiding to include or exclude studies without considering the inclusion and exclusion criteria. However, when it comes to the data analysis, there might still have been the possibility of incomplete findings or conclusions based on personal interest or opinions.

Avoiding misclassification of primary studies: as a measure to mitigate the risk of erroneous classification, it also validated the conferences and journals in which the papers were published, when applicable, as well as the references of each one, however, some misclassification may be occurred and it is not guaranteed that the outcome of the extraction made by other researches would be identical to those presented in this paper.

3.6 Study Limitations

Even considering English as being the most accepted language on academic publication, a diversity of studies not being published on this language can be observed, therefore not being considered in this systematic mapping.

The years considered as primary filter were between 2012 and 2019, however relevant studies associated with SOA and components could have been published before these years and were not considered on this systematic mapping.

4 Related Work

The authors Badampudi, Wohlin, and Petersen [10] conducted an SRL related to component-based software systems, which in accordance with them, requires decisions on component origins for acquiring components that are an alternative of where to get a component from. The SLR conducted by Badampudi, Wohlin, and Petersen [10] is similar to this present paper in terms of multiple criteria approach investigation, however, their paper did not include the MSA perspective or even the criticality aspect.

The authors [10] identified factors that could influence the decision to choose among different component origins and solutions for decision-making in the literature. In total was included twenty-four primary studies. The component origins compared were mainly focused on in-house compared with COTS (Components off-the-shelf) and COTS compared with OSS (Open Source Software). The authors identified eleven factors affecting or influencing the decision to select a component origin. Most of the solutions were proposed for in-house vs. COTS selection and time, cost and reliability were the most considered factors in the solutions.

Another SLR, conducted by Breivold, Crnkovic, and Larsson [11], evaluate the software evolution capability in the architecture perspective, intending to obtain an overview of the existing approaches in analyzing and improving software evolution capability at an architectural level, and investigate impacts on research and practice. The SLR conducted by Breivold, Crnkovic, and Larsson [11] is similar to this present paper in terms of the architectural perspective investigation, however, it did not include the criticality aspect.

In accordance with Breivold, Crnkovic, and Larsson [11], the software evolution capability describes a software system's ability to easily accommodate future changes, being a fundamental characteristic for making strategic decisions, and increasing economic value of software. For long-lived systems, there is a need to address evolution capability explicitly during the entire software lifecycle in order to prolong the productive lifetime of software systems. The authors identified five main categories of research topics, (i) techniques supporting quality consideration during software architecture design, (ii) architectural quality evaluation, (iii) economic valuation, (iv) architectural knowledge management, and (v) modeling techniques.

The findings of Breivold, Crnkovic, and Larsson's [11] review reveals that it is necessary to establish a theoretical foundation for software evolution research due to the fact that the expertise in this area is still built on the basis of case studies instead of generalized knowledge; as well as it is necessary to combine appropriate techniques to address the multifaceted perspectives of software evolution capability due to the fact that each technique has its specific focus and context for which it is appropriate in the entire software lifecycle.

The SLR conducted by Yingkui and Jing [12] evaluates the reliability analysis considering multiple possible states, which is known as multi-state (MS) reliability analysis. Multi-state reliability models provide more realistic and more precise representations of engineering systems, being more complex and present major difficulties in system definition and performance evaluation. This SRL present by presents Yingkui and Jing [12] is about the latest studies and advances about multi-state system reliability evaluation, multi-state systems optimization and multi-state systems maintenance, however, it's not associated with software architecture or MSA. This review is similar to this paper in terms of criticality associated.

There is still no general agreement for addressing the problem of criticality suggestions associated with MSA on literature. The analysis also reveals a broad spectrum of approaches supporting decision-making [10], architecture evaluation and review [11], and criticality [12], however, it was not possible to locate an SLR associated with all aspects that involve the selection methods for criticality in MSA.

5 Conclusions

This paper presented the process followed to perform an systematic mapping aiming to get an overview of existing researches on criticality suggestions related to MSA.

The research questions on this systematic mapping were intrinsically associated, considering that established selection models' question intends to identify models focused on helping the selection process associated to MSA, SOA and components in order to promote a base to decision making process.

The first research questions (RQ1) is directly associated with the multiple criteria evaluation approach question (RQ2), that aggregated with the criticality scope (RQ3), seeks to compose a well-grounded foundation of the most used criticality methods on the MSA perspective.

The results show an increasing interest in criticality research. They also suggest the immaturity of the area associated with MSA, since there is not an established guideline or proposal that focus on suggesting the critical microservices, differently from SOA and components that already have a diversity of studies around these areas.

Another weakness is the low number of works that performed comparative studies to highlight the advantages and disadvantages of their approaches and related work. However, this area has been showing signs of an increase in maturity since the numbers of published papers are increasing and MSA has been applied in practice.

A research trend was identified since the increasing number of experience reports in industry and academy showing evidence that the criticality approaches are feasible and beneficial to be applied in a real context. Nevertheless, further researches with more rigorous methods, including a case study for this purpose, would indicate new findings. Researchers intending to conduct researches related to criticality applied to MSA would use the results obtained by this paper to serve as a base to produce subsequent analysis in order to promote the maturity of this area. To address the lack of literature associated to criticality method related to MSA, the authors are conducting a research to propose a multiple criteria criticality method focused on MSA [19].

References

1. Australian National Health and Medical Research Council. How to review the evidence: systematic identification and review of the scientific literature, 2000. ISBN 186-4960329.
2. Australian National Health and Medical Research Council. How to use the evidence: assessment and application of scientific evidence. February 2000, ISBN 0 642 43295 2.
3. Cochrane Collaboration. Cochrane Reviewers' Handbook. Version 4.2.1. December 2003
4. Khan, Khalid, S., ter Riet, Gerben., Glanville, Julia., Sowden, Amanda, J. and Kleijnen, Jo. (eds) Undertaking Systematic Review of Research on Effectiveness. CRD's Guidance for those Carrying Out or Commissioning Reviews. CRD Report Number 4 (2nd Edition), NHS Centre for Reviews and Dissemination, University of York, IBSN 1 900640 20 1, March 2001.
5. D. D. de Carvalho, L. F. Chagas, A. M. Lima, and C. A. L. Reis, "Software process lines: A systematic literature review", in *International Conference on Software Process Improvement and Capability Determination*, Vilnius, Lithuania, 2014, p. 118–130.
6. O.P. Brereton, B.A. Kitchenham, D. Turner Budgen, M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain", Journal of Systems and Software Volume 80, Issue 4, April 2007, Pages 571-583
7. S. Haselböck, R. Weinreich, and G. Buchgeher, "Decision Models for Microservices: Design Areas, Stakeholders, Use Cases, and Requirements", in Software Architecture, ECSA 2017, Canterbury, UK, 2017, p. 155–170.
8. Lytra, I., Tran, H., Zdun, U.: "Supporting consistency between architectural design decisions and component models through reusable architectural knowledge transformations". In: Drira, K. (ed.) ECSA 2013, Montpellier, France, 2013. LNCS, vol. 7957, pp. 224–239. Springer, Heidelberg (2013). doi:10.1007/978-3-642-39031-9 20
9. Capilla, R., Jansen, A., Tang, A., Avgeriou, P., Babar, M.A.: "10 years of software architecture knowledge management: practice and future". Journal of Systems and Software Volume 116, June 2016, p. 191-205 (2015)

10. D. Badampudi, C. Wohlin, and K. Petersen, “Software component decision-making: In-house, OSS, COTS or outsourcing-A systematic literature review”, *Journal of Systems and Software*, vol. 121, p. 105–124, 2016.
11. H. P. Breivold, I. Crnkovic, and M. Larsson, “A systematic review of software architecture evolution research,” *Information and Software Technology*, vol. 54, no. 1, pp. 16–40, 2012.
12. G. Yingkui and L. Jing, “Multi-state system reliability: A new and systematic review,” *Procedia engineering*, vol. 29, pp. 531–536, 2012.
13. P. D. Francesco, I. Malavolta, and P. Lago, “Research on Architecting Microservices: Trends, Focus, and Potential for Industrial Adoption,” in 2017 IEEE International Conference on Software Architecture (ICSA), 2017, pp. 21–30.
14. N. Alshuqayran, N. Ali, and R. Evans, “A Systematic Mapping Study in Microservice Architecture,” in 2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA), Macau, China, 2016, pp. 44–51.
15. C. Pahl and P. Jamshidi. Microservices: A Systematic Mapping Study. In Proceedings of the 6th International Conference on Cloud Computing and Services Science, Rome, Italy, pages 137–146, 2016
16. F. Montesi and J. Weber, “Circuit Breakers, Discovery, and API Gateways in Microservices,” *CoRR*, vol. abs/1609.05830, 2016.
17. M. Gabbrielli, S. Giallorenzo, C. Guidi, J. Mauro, and F. Montesi, “Self-reconfiguring microservices,” in *Theory and Practice of Formal Methods*, Springer, 2016, pp. 194–210.
18. N. Dragoni et al., “Microservices: yesterday, today, and tomorrow,” CoRR, vol. abs/1606.04036, 2016.
19. E. F. M. de O. dos Santos and C. M. L. Werner, “A Survey on Microservices Criticality Attributes on Established Architectures,” in 2019 International Conference on Information Systems and Software Technologies, Quito, Equador, 2019, pp. 149–155
20. A. S. Agarwala, “Shortcomings in MIL-STD-1629A guidelines for criticality analysis,” in *Annual Proceedings on Reliability and Maintainability Symposium*, Los Angeles, CA, 1990, pp. 494–496.
21. J. Shao, X. Zhang, and Z. Cao, “Research on Context-based Instances Selection of Microservice,” in *Proceedings of the 2nd International Conference on Computer Science and Application Engineering*, Hohhot, China, 2018, pp. 1–5
22. J. Stubbs, W. Moreira, and R. Dooley, “Distributed Systems of Microservices Using Docker and Serfnode,” in *2015 7th International Workshop on Science Gateways*, Budapest, Hungary, 2015, pp. 34–39, doi: 10.1109/IWSG.2015.16.
23. I. Steinmacher, A. P. Chaves, and M. A. Gerosa, “Awareness support in distributed software development: A systematic review and mapping of the literature,” Computer Supported Cooperative Work (CSCW), vol. 22, no. 2-3, 2013, pp. 113–158.