International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 15, No.1 (Mar-24)

http://dx.doi.org/10.12785/ijcds/1501100

Microservices Adoption: An Industrial Inquiry into Factors Influencing Decisions and Implementation Strategies

Mehdi AIT SAID[1], Abdellah EZZATI[1], Soukaina MIHI[1] and Lahcen BELOUADDANE[1]

[1]*Hassan First University, Settat, Morocco*

Received 28 Dec. 2023, Revised 10 Mar. 2024, Accepted 11 Mar. 2024, Published 15 Mar. 2024

Abstract: Microservices Architecture (MSA) has emerged as a promising paradigm for building scalable and flexible software systems. While extensive research focuses on MSA's technical aspects, there remains a gap in understanding how practitioners make decisions to adopt and implement MSA in real-world organizational contexts. To address this gap, we conducted an in-depth qualitative study through 30 semi-structured interviews with experienced practitioners in the field. Our investigation unveils the intricate factors driving practitioners' decision-making processes during MSA adoption. We highlight the multifaceted influences of reusability, scalability, extensibility, maintainability, and other factors, shedding light on the motivations behind adopting MSA. Moreover, we delve into key strategies practitioners employ during MSA adoption, emphasizing the importance of the Modulith approach as a bridge between monolithic and MSA Our findings underscore the significance of practitioner experience in shaping MSA adoption decisions.

Keywords: Microservices Architecture, Adoption, Decision-Making, Modulith, Industrial Inquiry, Qualitative Study

1. Introduction

In the rapidly evolving realm of software architecture, Microservices Architecture (MSA) has emerged as a groundbreaking paradigm that promises to revolutionize the way modern applications are designed, developed, and deployed [1]. Unlike traditional monolithic architectures, where applications are built as large, interconnected entities, MSA advocates for the decomposition of complex systems into smaller, independent services known as microservices [2]. Each microservice encapsulates a specific business functionality and communicates with others through well-defined interfaces [3]. The benefits of adopting MSA are manifold and far-reaching. One of the most significant advantages lies in its inherent modularity, which promotes agility and scalability [4]. With MSA, development teams can work on individual microservices independently, enabling faster development cycles and allowing for swift changes without disrupting the entire system [5][6]. Scalability is another key feature; as microservices can be scaled individually based on their specific resource requirements, the architecture is well-suited to handle varying workloads and accommodate growth seamlessly [7][8]. Furthermore, MSA enhances fault isolation. Since each microservice operates independently, a failure in one service does not necessarily impact others, resulting in increased system reliability and resilience [9]. This architecture also aligns well with modern development practices like DevOps and Continuous Integration/Continuous Deployment (CI/CD), facilitating faster and more frequent releases [10][11]. In addition to technical advantages, MSA brings about organizational benefits. It enables cross-functional teams to work on individual microservices, promoting a decentralized development approach and empowering teams to take ownership of their services [12]. This fosters a culture of accountability and innovation within development teams. As organizations strive to stay competitive in the digital age, MSA offers a strategic advantage. Its ability to accommodate evolving business requirements, respond to changing market demands, and deliver software faster makes it a prime choice for companies seeking to achieve digital transformation and maintain a competitive edge [13].

However, while MSA offers a multitude of advantages, its implementation is not devoid of challenges. The very modularity that fosters agility can introduce complexities in terms of communication between microservices, data consistency, and overall system monitoring [14]. Orchestrating the interactions between these granular components demands careful consideration to avoid a fragmented ecosystem [15]. Additionally, ensuring proper testing and deployment across numerous services can become intricate [16]. It's important to note that MSA might not be suitable for all projects and organizations, as the decision to

E-mail address: m.aitsaid@uhp.ac.ma, abdellah.ezzati@uhp.ac.ma, soukaina.mihi@uhp.ac.ma, lahcen.belouaddane@uhp.ac.ma

adopt it should be based on factors such as project scope [17], team expertise, and existing infrastructure. Therefore, understanding and effectively addressing the inherent complexities of MSA adoption is crucial for its successful integration within diverse organizational contexts [18][19].

The intricacies of MSA adoption, the factors that drive it, and the practical strategies for its successful integration have remained subjects of interest and exploration. In light of this, our motivation for this study stems from a profound need to bridge the gap between the theoretical understanding of MSA adoption and its real-world implementation in the industry. A notable catalyst for our research emerged from an observation in March 2023, when Amazon made headlines by announcing their transition from a distributed MSA to a monolith application, yielding substantial benefits such as increased scale, enhanced resilience, and an astonishing 90% cost reduction in Prime Video's audio/video monitoring [20]. This revelation sparked our curiosity, highlighting a divergence between conventional wisdom about MSA adoption and the pragmatic choices made by industry leaders. Against this backdrop, our research embarks on a journey to address this disparity and uncover the underlying dynamics that shape the adoption of MSA. To this end, we have set two primary objectives:

- **Our first objective** involves delving into the intricate landscape of factors influencing practitioners' decisions when adopting MSA. By investigating these influences, we aim to gain a nuanced understanding of the motivations, triggers, and considerations that prompt organizations to embrace MSA.
- **Our second objective** focuses on revealing the key practices and strategies employed by practitioners during the adoption of MSA within their projects. Through this inquiry, we aim to uncover practical insights, approaches, and methodologies that guide practitioners in successfully transitioning from traditional monolithic architectures to the intricate landscape of MSA.

Our study is propelled by the aspiration to update the state-of-practices of the adoption of MSA and to bridge the gap between MSA theory and its real-world adoption. By addressing these objectives, we aim to contribute to the practical understanding of MSA adoption and empower practitioners with actionable insights that facilitate informed decision-making and successful integration of MSA within diverse organizational contexts. To achieve this goal, we employed an industrial inquiry and qualitative research methodology. This involved conducting a comprehensive series of semi-structured interviews with 30 practitioners deeply engaged in the field of MSA.

This study contributes significantly to MSA adoption understanding and practice. It reveals the multifaceted factors guiding practitioners' decisions, providing insights into motivations and contextual considerations. Additionally, it details key practices, offering a practical framework for successful adoption. The study advances understanding of the Modulith strategy as an effective intermediate approach, combining monolithic simplicity with microservices' modularity. Lastly, it emphasizes the need for further research, particularly in standardizing migration and decomposition strategies, and underscores the importance of practitioner experience in MSA adoption decisions.

The remainder of this paper is structured as follows: Section 2 provides an exploration of the existing literature in the field of MSA. In Section 3, we examine related works that have contributed to the understanding of MSA adoption. Section 4 outlines the research methodology adopted for this study, detailing our approach to data collection and analysis. In Section 5, we present the demographic results of the practitioners who participated in our study. Section 6 delves into the findings that emerged from our investigation into MSA adoption practices. Subsequently, Section 7 engages in a comprehensive discussion of the implications and insights drawn from the findings. Section 8 highlights potential threats to the validity of our study's conclusions. Finally, Section 9 concludes the paper by summarizing the main takeaways and suggesting possible avenues for future research.

2. Literature review

The literature review in the field of MSA has primarily centered on several key areas, including state-of-the-art exploration [21][22][23][24], design patterns detection [25][26][27], migration and decomposition methods [28][29], and addressing challenges associated with MSA [30]. A substantial body of research has delved into understanding the theoretical foundations, architectural principles, and technological advancements that underpin the MSA paradigm. Moreover, studies have extensively investigated design patterns that aid in structuring and organizing microservices to achieve optimal system performance, scalability, and maintainability.

A notable portion of research efforts has been directed toward the development of effective migration and decomposition strategies aiming to facilitate the transition from traditional monolithic architectures to MSA such as [28][29][31]. These strategies often emphasize architectural transformations, tooling support, and best practices to ensure a successful migration process. Furthermore, the literature has been attentive to various challenges posed by MSA adoption, including issues related to communication, data consistency, monitoring, security, and fault tolerance. These studies such as Söylemez et al. [30] and Munaf et al. [32] have provided valuable insights into mitigating obstacles and optimizing the benefits of MSA implementation.

However, while the existing body of work has significantly contributed to the theoretical understanding of MSA, there remains a gap in understanding how practitioners in real-world organizational contexts make decisions regarding

MSA adoption and how they navigate the intricacies of its implementation. This study seeks to address this gap by adopting an industrial inquiry and qualitative research approach, centered on practitioner perspectives and experiences. By focusing on the decision-making processes, factors influencing adoption choices, and the practical strategies employed by practitioners, this research aims to provide a holistic understanding of the intricacies and dynamics surrounding MSA adoption. In doing so, our study contributes to bridging the divide between theoretical knowledge and real-world practices, offering valuable insights that can inform practitioners, researchers, and organizations seeking to embark on successful MSA adoption journeys.

3. Related work

In 2022 Zhou et al [33] conducted an industrial inquiry that sheds light on the real-world adoption of MSA through an empirical lens. This investigation aims to uncover the gaps between idealized visions of MSA presented by Fowler and Lewis in 2014 [3] and the actual industrial practices, as well as the challenges practitioners face during adoption. The authors employ a qualitative research approach involving semi-structured interviews with practitioners from 20 software companies in China. Their findings highlight a set of common practices and associated challenges in the industry, providing insights into the complexities of MSA. Furthermore, they identify key decision points that require careful consideration by practitioners to balance the benefits and challenges of MSA.

Comparing this study with our own, we note a similar focus on empirically exploring MSA adoption in real-world contexts. However, our study diverges in its objectives, as we aim to delve into the factors influencing practitioners' decisions to adopt MSA and the key practices and strategies they employ during the adoption process, our research seeks to uncover the motivations and triggers for MSA adoption and provide a comprehensive understanding of how practitioners navigate the adoption journey. While the study conducted by Zhou et al [33]. sheds light on the gaps between the ideal characteristics presented by Fowler and Lewis [3] and the realistic practices of MSA. Our study provides a more focused and detailed examination than the Zhou et al work [33]. We delve deeply into the decision-making processes behind MSA adoption and meticulously explore the strategies practitioners employ during adoption. By thoroughly analyzing both the decision-making factors and the intricate implementation strategies, our research contributes to a holistic understanding of MSA adoption practices and provides a comprehensive framework for practitioners seeking guidance in their adoption journey.

The survey conducted by Velepucha and Flores [16] offers a comprehensive exploration of MSA, emphasizing principles, patterns, and migration challenges. They concentrate on providing a detailed explanation of monolithic and MSA, highlighting foundational software engineering concepts. Their work presents patterns for microservices design and emphasizes the importance of Domain-Driven Design (DDD) for replaceability and autonomy. In contrast, our research delves into real-world practitioner perspectives, focusing on the decision-making factors and strategies employed during MSA adoption. Both studies contribute valuable insights to the understanding of MSA, yet the approaches differ significantly. Velepucha and Flores [16] offer a theoretical foundation and recommend DDD for successful migrations, providing a roadmap for researchers and professionals. In comparison, our study takes a qualitative research approach, exploring the motivations and triggers behind practitioners' decisions to adopt MSA, providing a more in-depth and nuanced understanding of the adoption process. The studies, when considered together, offer a complementary perspective on MSA adoption, combining theoretical principles with practical insights from practitioners, enriching the overall understanding of this complex transition.

4. Research method

As illustrated in Figure 1, our study follows an industrial inquiry and qualitative research approach, seeking to provide deep insights into the intricacies of MSA adoption. To achieve this, we conducted a comprehensive set of interviews with 30 practitioners deeply entrenched in the realm of MSA. The commencement of this study took place in April 2023, extending over a duration of three months. The research methodology employed aligned with a well-established qualitative approach, namely interviews as delineated by Brinkmann & Kvale [34]. The collaborative efforts of the three Ph.D. student researchers and their supervisor were instrumental in devising the interview framework, orchestrating data collection by engaging with practitioners through interviews, and subsequently undertaking data analysis to facilitate comprehensive reporting.

A. *Research questions*

This study seeks to comprehensively explore the realm of MSA adoption within organizational contexts. To achieve this, two key research questions were formulated, each with a distinct objective aimed at unraveling different aspects of practitioners' experiences with MSA adoption.

1) *RQ1: What factors influence practitioners' decision-making processes in adopting MSA in their organizations?*

The primary objective of RQ1 is to delve into the intricate landscape of factors that shape practitioners' decision-making processes when adopting MSA. By investigating this question, we aim to gain insights into the underlying motivations, triggers, and considerations that prompt organizations to embrace MSA. The exploration of these factors will enable us to unravel the nuanced interplay between the decision to adopt MSA and the contextual dynamics of the organizations in which it is implemented. Furthermore, understanding these influences will contribute to a more comprehensive understanding of the adoption journey, shedding light on why and when MSA is chosen as an architectural paradigm.

2) RQ2: What are the key practices and strategies employed by practitioners during the adoption of MSA in their projects?

The focal point of RQ2 is to illuminate the key practices and strategies that practitioners employ as they navigate the adoption of MSA within their projects. Through this inquiry, we aim to uncover the practical insights, approaches, and methodologies that guide practitioners in successfully transitioning from traditional monolithic architectures to the intricate landscape of MSA. By investigating the strategies used, the sequence of steps taken, and the underlying principles adhered to, we intend to provide a comprehensive framework that practitioners and organizations can draw upon when embarking on their own MSA adoption journeys. Ultimately, the objective of this research question is to contribute to a knowledge base that enhances the practical understanding of MSA adoption and equips practitioners with valuable guidance for its effective implementation.

B. Research procedure

The research procedure for this study was meticulously designed to ensure a systematic and comprehensive exploration of MSA adoption. The interviews were conducted with a substantial emphasis on fostering comprehensive discussions between the interviewees and the interviewers. This approach entailed a combination of open-ended questions, strategically designed to elicit both anticipated and unforeseen insights throughout the interview process. This dynamic approach to data collection facilitated a comprehensive spectrum of information to be gleaned from the interviews, as indicated in the structured questionnaires provided in Table I. The procedure encompassed five stages, each aimed at gathering insights from experienced practitioners in the field.

- **Brainstorming and Interview Plan Development:** The initial stage involved collaborative brainstorming among the four researchers of this work to identify key areas of interest related to MSA adoption. This collective effort allowed for the delineation of pertinent topics and themes that would guide the subsequent interview process. Through this process, the researchers aimed to ensure the exploration of diverse aspects of MSA adoption, thereby enhancing the comprehensiveness of the study.

- **Design of Interview Instrument:** Building upon the outcomes of the brainstorming sessions and drawing inspiration from previous MSA-related survey studies, the researchers proceeded to design the interview instrument. This instrument was meticulously crafted to align with the identified areas of interest and aimed to elicit in-depth insights from practitioners. The questions within the instrument were thoughtfully formulated to facilitate a rich and nuanced exploration of MSA adoption experiences.

- **Practitioner Selection and Invitation:** Concurrently, the researchers embarked on the task of identifying suitable practitioners to participate in the interviews. To ensure a diverse pool of experienced practitioners, the researchers focused on participants from 3 international software engineering conferences and tow tech enterprises expos held in Morocco and France. These conferences and expos were recognized for attracting experts in the field, making them ideal platforms for practitioner recruitment.

- **Criteria-Based Selection:** The selected practitioners were those with a substantial background in the domain of software architecture, having accumulated more than five years of industry experience in software engineering, including at least three years of practical engagement with MSA. This meticulous criterion aimed to ensure that the participating practitioners possessed substantial insights and firsthand experience to contribute to the study's objectives.

- **Interview Initiation:** The final stage of the research procedure commenced with the initiation of interviews with practitioners who accepted the invitation. The researchers engaged in open and exploratory conversations with the selected practitioners, utilizing the interview instrument as a guide. These interviews aimed to uncover practitioners' perspectives, experiences, and insights related to MSA adoption.

TABLE I. The interview questions

No.	Question
Q1	How many years of experience do you have working with MSA? Can you briefly describe your role and involvement in projects related to MSA during that time?
Q2	Can you describe a specific instance when your organization decided to adopt MSA?
Q3	How did MSA characteristics play a role in your decision to adopt MSA?
Q4	What challenges or concerns did you encounter during the decision-making process, and how did you address them? Were there any specific stakeholders or internal dynamics that influenced the decision?
Q5	What specific practices and strategies did you employ during the adoption of MSA in your projects?
Q6	What were the major challenges you faced during the adoption of MSA? Can you share the specific strategies or approaches you used to overcome these challenges and ensure successful adoption?
Q7	What are the factors that you considered in adopting MSA?

C. Data collection

The data collection process for this study employed a semi-structured interview approach, incorporating a blend of straightforward open-ended inquiries to comprehensively capture anticipated and unanticipated data across interviews. To accommodate the language preferences of the practitioners, interviews were conducted in Arabic, French, or English, languages proficiently understood by the interviewers. The interviews were primarily conducted face-to-face, undertaken at the interviewees' locations spanning nine cities across Morocco and France. However, geographic constraints necessitated that eight interviews be conducted online. The data collection phase extended over a span of approximately three months, encompassing a total of 30 interviews. Each interview session involved the participation of one interviewee and four researchers functioning as interviewers.

During the interviews, one researcher facilitated the question-and-answer process guided by the interview instruments, while the remaining researchers assumed note-taking responsibilities and posed supplementary queries to ensure comprehensive coverage of the topics of interest. The entirety of the interview sessions accumulated to a total duration of 27 hours and 35 mins, with an average duration of 55 minutes per interview. As a precaution against any loss of significant insights, all interviews were recorded using a digital voice recorder following the explicit consent of the interviewees. This meticulous approach aimed to ensure the accuracy and thoroughness of the collected data.

D. Data synthesis

The process of data synthesis encompassed strict meticulous steps to ensure a comprehensive analysis of the interview data. Initially, a researcher meticulously transcribed the interview recordings, followed by a rigorous dual verification conducted by the other two researchers to uphold accuracy. To mitigate the potential influence of researcher bias, a strategy of independent coding was implemented. Each interview transcript was coded by the three researchers individually, contributing to a multifaceted and unbiased perspective in the analysis process. Adhering to the guidance of Welsh [35], the data synthesis procedure employed a blend of manual and computer-assisted techniques. The qualitative data synthesis tool, NVivo[1] , was leveraged to facilitate efficient pattern recognition within substantial volumes of textual data, enhancing the precision of the synthesis process. Subsequently, the data underwent thematic synthesis, a method outlined by Patton [36]. This method involved a progressive encoding process applied to the data. The researchers conducted a systematic manual analysis to identify prominent themes, concerns, and practices associated with various phases of MSA adoption.

5. Demographic results

This section presents the demographic characteristics of the interviewees, providing valuable insights into the profiles of the participants who contributed to the study. Table II provides a concise overview of the demographic attributes of the interviewees, encompassing their respective domains, locations, and organization sizes, these demographic insights contribute to the robustness and applicability of the study's findings, reflecting a diverse cross-section of professionals engaged in MSA adoption across distinct organizational contexts.

Professional Roles: The interviewees represent a spectrum of roles within the MSA domain. The distribution of roles includes project managers (8/30), senior software architects (18/30), and tech leads (4/30). This diverse mix of roles reflects the broad engagement of professionals at varying levels of responsibility and expertise within the MSA landscape.

Experience and Adoption: Each of the interviewed practitioners brings substantial experience to the study. With a minimum of five years in software engineering, these individuals possess a wealth of knowledge in the software development field. Significantly, every practitioner has dedicated a minimum of three years to adopting MSA. This depth of experience underscores the participants' familiarity with MSA principles, practices, and challenges.

Domain, Location, and Organization Size: The demographic diversity extends to the domains of the interviewees' organizations. Participants hail from six domains, Technology and Software Development (S), Telecommunications (T), E-commerce (E), Business Management (E), IT Service Provision (P), and Banking and Finance (BF). This assortment of industries enriches the study's applicability across sectors. Additionally, the participants are dispersed across various cities, including Casablanca, Paris, Lyon, and Rabat, spanning Morocco and France. Moreover, the size of the organizations represented varies, from local businesses with fewer than 100 employees to multinational corporations boasting more than 10,000 personnel such as IBM and Oracle. This range in organization size encapsulates the broad spectrum of enterprises engaging with MSA.

[1] https://lumivero.com/products/nvivo/

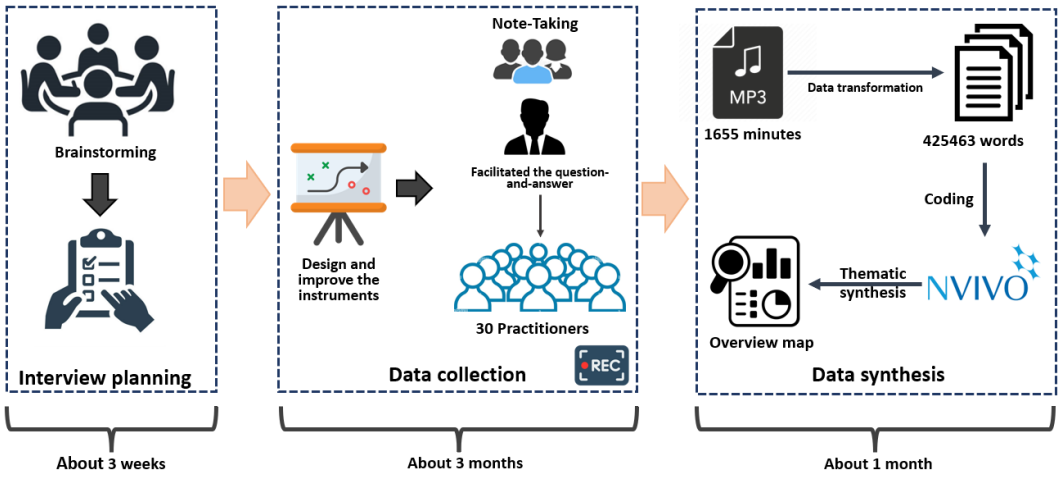

Figure 1. The procedure of the interview study

TABLE II. The detailed information of the 30 practitioners

No.	Domain	Company Location	Company Size
S1	S	Multinational	>10,000
S2	S	Multinational	>10,000
S3	S	Casablanca/Morocco	<100
S4	S	Casablanca/Morocco	<100
S5	S	Paris/France	5000–10,000
S6	S	Lyon/France	1000–5000
T1	T	Multinational	5000–10,000
T2	T	Multinational	5000–10,000
T3	T	Casablanca/Morocco	5000–10,000
E1	E	Casablanca/Morocco	100–1000
E2	E	Casablanca/Morocco	100–1000
E3	E	Casablanca/Morocco	1000–5000
E4	E	Marrakech/Morocco	<100
B1	B	Casablanca/Morocco	100–1000
B2	B	Casablanca/Morocco	100–1000
B3	B	Casablanca/Morocco	1000–5000
B4	B	Casablanca/Morocco	1000–5000
B5	B	Casablanca/Morocco	1000–5000
B6	B	Multinational	100–1000
B7	B	Paris/France	1000–5000
B8	B	Paris/France	1000–5000
B9	B	Rabat/Morocco	100–1000
P1	P	Multinational	>10,000
P2	P	Multinational	>10,000
P3	P	Casablanca/Morocco	100–1000
P4	P	Casablanca/Morocco	100–1000
P5	P	Casablanca/Morocco	1000–5000
BF1	BF	Casablanca/Morocco	100–1000
BF2	BF	Casablanca/Morocco	5000–10,000
BF3	BF	Rabat/Morocco	100–1000

6. Findings

The focus of this study is to provide a comprehensive overview of the contemporary practices in the adoption of MSA. The primary objective is to gain insights into the decision-making processes surrounding the utilization of MSA within diverse industries. This research seeks to elucidate the specific circumstances that prompt organizations to embrace MSA, exploring both the timing and strategies involved in this transition. Additionally, the study aims to delve into the nuances of how MSA is adopted, shedding light on the various approaches employed by industry practitioners. By examining the "when" and "how" of MSA adoption, this research strives to contribute valuable insights into the current landscape of architectural transformations.

A. Time to Use

This section explores the specific instances and reasons that led practitioners to adopt MSA extracted from Q2. Through analyzing the responses from the practitioners, it becomes evident that the timing of MSA adoption varied across different organizations. The graph in Figure 2 presents the seven detected influences on the adoption of MSA across six domains.

1) Early adoption for scalability and agility

Among the respondents, 23 out of 30 practitioners adopted MSA early in their projects to address scalability and agility concerns. Recognizing the need for a more flexible and responsive architecture, they sought to accommodate future growth and efficiently handle varying workloads. MSA provided a solution by breaking down monolithic applications into independent microservices, allowing granular scalability, optimized resource utilization, and faster development cycles [7][37]. This strategic choice afforded organizations a competitive advantage, improving time-to-market for products and services and enhancing adaptability to dynamic market demands.

2) Reactive adoption due to monolithic challenges

For 13 practitioners, the adoption of MSA was reactive and driven by challenges encountered with their existing monolithic architecture. Cumbersome maintenance, longer

development cycles, and the inability to scale specific features independently were key challenges. The limitations of monolithic architectures prompted these organizations to consider MSA as a solution to address these issues. MSA adoption provided a remedy by introducing modularity, scalability, and agility, allowing organizations to overcome the limitations of their monolithic systems.

3) Strategic long-term planning

A smaller subset of practitioners (4 out of 30) strategically adopted MSA as part of their long-term planning. Recognizing the potential benefits early on, these organizations aligned their architecture to be more modular and flexible from the beginning. They foresaw the increasing complexity of their applications and wanted a modular architecture that could accommodate future expansions and seamlessly integrate new services. MSA adoption was viewed as a strategic decision to build a scalable and adaptable system, providing a solid foundation for future innovations.

4) Post-market release and customer demands

Feedback from customers and their demands for frequent updates influenced the adoption of MSA for 14 practitioners. The decision came after launching products or services to the market, where organizations received valuable feedback. Customers in the fast-paced digital landscape expect regular updates and continuous improvement. MSA allowed organizations to deploy updates more frequently and respond to customer needs faster. The modular nature of microservices facilitated independent development, testing, and deployment, enabling organizations to meet customer demands more effectively.

5) Legacy system modernization

A significant number of practitioners (19 out of 30) adopted MSA as part of their legacy system modernization efforts. Facing challenges with aging monolithic systems accumulating technical debt and limited scalability, these organizations sought a solution to modernize their applications and make them more responsive to business changes. MSA offered a way to address these challenges by providing modularity, scalability, and adaptability, allowing organizations to overcome the limitations of their aging monolithic systems.

6) Tech stack updates and adoption of cloud services

The decision to adopt MSA was influenced by technological advancements and the increasing adoption of cloud services for 17 practitioners. Updating technology stacks and migrating infrastructure to the cloud led these organizations to recognize the potential benefits of cloud-native architectures, with MSA emerging as a natural fit. MSA aligned well with cloud-native capabilities, offering scalability, resilience, cost savings, faster innovation cycles, easier management, and maintenance. Cloud services played a significant role in shaping the decision to transition to MSA.

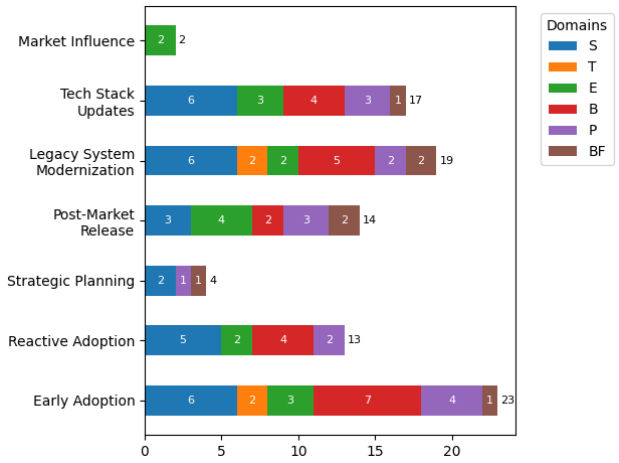

Figure 2. Influences on the adoption of MSA across different domains

7) Market influence and industry trends

Two respondents from the E domain mentioned that market influence and industry trends played a role in their decision to adopt MSA. Observing the increasing adoption of MSA by leading organizations in their industry, these practitioners recognized the potential benefits of embracing a microservices-based approach. They decided to follow suit to stay relevant, meet customer expectations, and gain a competitive edge. Industry leaders often set the tone for best practices and innovation, making the adoption of MSA an opportunity to align with industry standards and proven approaches to software development. Here is an interesting quote from E3 that illustrates this market influence: ”MSA gained significant traction in our industry, and we noticed that many successful companies were adopting it. We saw the potential to improve our development cycles and respond better to customer needs. It made strategic sense to follow the industry trend and leverage the advantages of MSA.”

B. Decision-making factors

Decision-making factors are crucial considerations that influence the adoption of MSA across different domains. Based on responses both Q3 and Q4 from practitioners in various industries, the Figure 3 presents the frequency of mentions for each factor from the 8 found factors, shedding light on the factors that play a significant role in the decision-making process.

1) Reusability

Reusability stands out as a critical decision-making factor, consistently mentioned across all domains with a total of 17 mentions. Practitioners recognized the strategic value of developing microservices that could be reused across different projects, leading to substantial time and effort savings in the long run. By focusing on modularity and reusability in their design approach, organizations aimed to create a repository of reusable components, fostering

code sharing and standardization. This strategic adoption of MSA underscores the commitment to building a sustainable foundation for future development projects.

2) Scalability

Scalability emerged as a pivotal factor driving MSA adoption across various domains, accumulating a total of 23 mentions. The need for a system capable of handling increased workloads and adapting to changing market demands drove practitioners to adopt MSA. Breaking down monolithic applications into independent microservices allowed organizations to scale each service independently based on demand. This approach ensured optimal resource utilization, contributing to improved performance. Scalability played a central role in the decision to embrace MSA, enabling dynamic responses to evolving business requirements and market demands.

3) Extensibility

Extensibility garnered 21 mentions as a significant factor influencing MSA adoption. Organizations recognized the importance of building a system that could easily accommodate changing business requirements and allow seamless modifications and extensions. Embracing principles of loose coupling and employing modular design patterns, practitioners ensured that each service could be extended or modified without impacting other parts of the system. This emphasis on extensibility supported faster and more efficient development cycles, enabling organizations to stay relevant and adapt to evolving business needs.

4) Maintainability

Maintainability emerged as a critical factor with 24 mentions in the decision-making process for adopting MSA. Practitioners emphasized the importance of building a system that is easy to maintain and update over time. By adhering to clean code practices, comprehensive documentation, and implementing automated testing and continuous integration, organizations ensured that their microservices remained maintainable and could be iterated upon efficiently. This commitment to maintainability reflects a strategic approach to reducing technical debt and ensuring system stability in the face of evolving requirements.

5) Technology heterogeneity

While not as prominently mentioned, Technology Heterogeneity received 9 mentions, particularly in the S, E, and P domains. Organizations in these domains recognized the relevance of seamlessly integrating diverse technologies to achieve business objectives effectively. Adopting MSA allowed them to develop microservices using different technologies, facilitating efficient communication through standardized APIs. This strategic approach addressed the challenges of integrating multiple technologies, ensuring smooth data exchange and functionality across the system.

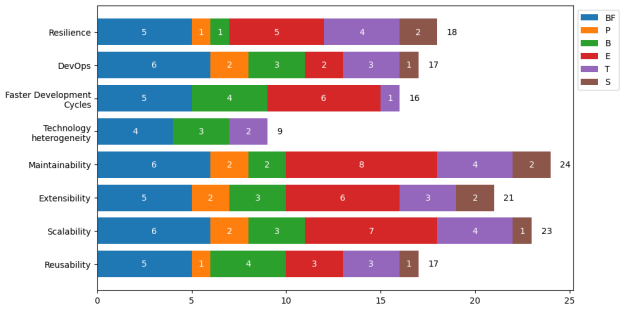

Figure 3. Influences frequency of key factors mentioned by domain

6) Faster development cycles

Faster Development Cycles emerged as an essential consideration, accumulating 16 mentions in the decision-making process for adopting MSA. Organizations acknowledged the significance of accelerating development processes to respond quickly to market demands and stay competitive. MSA facilitated rapid development by breaking down monolithic applications into manageable microservices. Each microservice could be developed independently, allowing for parallel development and deployment, resulting in reduced time-to-market for new products and services.

7) DevOps

DevOps played a crucial role in the decision-making process for adopting MSA, receiving a total of 17 mentions. Organizations recognized the need to streamline development and operations processes for better collaboration and faster deployment cycles. The implementation of DevOps practices in conjunction with MSA enabled seamless communication and collaboration between development, testing, and operations teams. This approach fostered automation and CI/CD, allowing for faster and more reliable software deployments, enhancing overall organizational agility and responsiveness.

8) Resilience

Resilience emerged as a critical consideration with 18 mentions in the decision-making process for adopting MSA. Organizations emphasized the importance of building a system that could withstand failures and maintain overall availability. Through fault tolerance mechanisms and distributed architectures, practitioners built resilient systems capable of handling failures gracefully and recovering quickly. This strategic adoption of MSA ensured consistent and reliable system performance, even in the face of unexpected events.

C. Adoption strategies

The practitioners employed two primary adoption strategies for transitioning to MSA: the Strangler strategy and the Demolition strategy. The extracted resalt of Q5 and Q6 is presented in Figure 4, out of the total 30 respondents, 24 practitioners opted for the Strangler strategy, while 6 practitioners chose the Demolition strategy.

1) Strangler strategy

The Strangler strategy, which garnered significant adoption by 24 practitioners, proved to be a favored approach in transitioning from existing monolithic systems to MSA. The practitioners from different domains highlighted its advantages, stating that they chose the Strangler strategy because it allowed for a gradual migration. By gradually replacing specific functionalities with microservices, they minimized the risk of disruption and ensured a smooth transition, contributing to a more manageable and successful adoption. Within the Strangler strategy, practitioners employed various Decomposition Methodologies to guide their transition:

- **Domain-Driven Design:** 14 practitioners opted for DDD to guide their shift to MSA. DDD involves breaking down monolithic applications along domain boundaries, assigning each microservice specific business capabilities [38]. This aligns architecture with business logic, fostering a modular and cohesive design. DDD enhances application understanding, promotes isolated microservices for easier management, and enables scalable, domain-specific services for optimal resource utilization [39]. The approach accelerates development cycles, allowing concurrent work on microservices and swift deployment. A practitioner from the B domain emphasized DDD's effectiveness, stating, "We chose DDD because we wanted to align our application with our business logic. By breaking down the monolithic application into domain-specific services, we achieved a more organized and maintainable architecture. It also helped us scale individual services based on demand, ensuring optimal performance..." (B5).

- **Experience-Driven Decomposition (EDD):** 5 practitioners drew from past project experiences to guide their transition to MSA. By learning from prior endeavors, they anticipated challenges, applied best practices, and avoided past mistakes, streamlining the migration process. Leveraging insights from similar projects enabled them to foresee hurdles, strategize effectively, set realistic timelines, and allocate resources wisely, ensuring a smoother transition. A practitioner from the S domain highlighted their strategic approach, stating, "Drawing on our experience from similar projects allowed us to approach the migration strategically. We were able to identify potential pitfalls early on and devise effective solutions. It gave us confidence in our decisions and ensured a smooth transition" (S3).

- **Functional Decomposition:** 3 practitioners embraced the functional decomposition methodology to transition to MSA. This method involves breaking down the monolithic application into smaller, independent functional components, enhancing modularity and flexibility. By isolating specific functionalities into standalone microservices, teams can work independently on each service, improving development efficiency and facilitating maintenance and updates. The granularity of this approach promotes reusability, reducing redundant efforts and fostering code sharing across projects [40]. This strategy leads to optimized resource utilization, faster development cycles, and a more organized application structure. A practitioner from the P domain affirmed the positive impact, stating, "Functional decomposition was a game-changer for our development process. It allowed us to build smaller, focused services that we could easily reuse across multiple projects. This approach made our codebase cleaner and more maintainable, and it significantly improved our development speed..."(P5).

- **Data-Driven Decomposition:** 2 practitioners chose Data-Driven decomposition for their transition to MSA. This approach organizes microservices around specific data entities, improving data handling efficiency, reducing the risk of inconsistencies, and enhancing scalability [28]. By aligning microservices with data entities, practitioners achieved better data encapsulation and streamlined architecture [41]. A practitioner from the BF domain stated, "Data-driven decomposition played a critical role in our successful transition to MSA. By organizing our services around data entities, we improved data handling efficiency and reduced the risk of data inconsistencies. Our microservices now handle specific data-related tasks effectively, making our architecture more resilient and easier to scale…" (BF2).

2) Demolition strategy

The Demolition strategy was embraced by 6 practitioners as their preferred approach to adopting MSA. One practitioner from the T domain shared their perspective, stating, "Our decision to adopt the Demolition strategy was driven by the need for a complete overhaul of our legacy system. We decided to dismantle the monolithic application and build a new system entirely based on microservices to address our scalability and maintainability concerns." (T1).

The demolition strategy is a radical approach that involves replacing the entire monolithic application with a new architecture based on microservices. Unlike the strangler strategy, which entails gradual migration, the Demolition strategy advocates for a complete removal of the existing system and a fresh start with microservices. Practitioners who adopt the demolition strategy recognize that their legacy system may have significant technical debt and scalability limitations that are difficult to address incrementally. Instead of incrementally replacing components, they prefer to build a new microservices-based architecture from scratch to ensure a clean slate and a more robust foundation for future growth.

In adopting the Demolition strategy, 6 practitioners utilized 2 decomposition methodologies for their transition

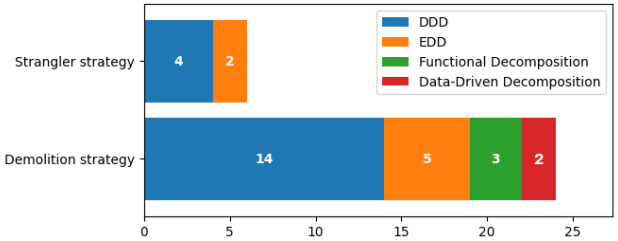

Figure 4. Decomposition methodologies for strangler and demolition strategies

to a microservices-based system. 4 practitioners employed DDD to ensure clear domain boundaries, fostering a focused and maintainable architecture. Meanwhile, 2 practitioners opted for an EDD, leveraging insights from past projects to identify best practices and navigate challenges during the development process.

3) Decomposition factors

In the process of decomposing monolithic systems into MSA, practitioners consider multiple factors, as depicted in Figure 5 , with these findings extracted from Q7. Reusability, highlighted with 25 mentions, emerges as pivotal, emphasizing the creation of modular microservices for future reuse. Scalability, with 24 mentions, underlines the importance of independently scaling services based on demand. Extensibility, cited 23 times, is crucial for creating adaptable microservices that facilitate seamless incorporation of new functionalities. Maintainability, mentioned 25 times, is critical, ensuring ease of management, updates, and troubleshooting of individual microservices. Resilience, noted 16 times, focuses on enhancing system robustness against failures. Deployment and Infrastructure, with 19 mentions, is vital for efficient, resilient deployment processes, often involving containerization and orchestration technologies. Team Structures and Autonomy, cited 21 times, emphasizes decentralization and dedicated teams for independently deployable microservices. Data Ownership, mentioned 19 times, stresses the importance of assigning clear data ownership to microservices for better consistency and integrity. Testing, with 27 mentions, becomes paramount for validating individual microservices' functionality, reliability, and interactions. Monitoring, noted 26 times, plays a crucial role in ensuring the health, performance, and reliability of microservices and the entire system. Lastly, Communication, mentioned 23 times, is vital for coordinating interactions between microservices, with a focus on effective communication protocols and well-defined APIs. The practitioners' perspectives highlight the strategic considerations that guide the decomposition process and contribute to the successful adoption of MSA.

4) Modulith architecture

The Modulith architecture, a fusion of "Module" and "Monolith," presents a strategic approach to application structuring, positioning itself between traditional Monolithic and MSA. In a Monolithic context, all functionalities reside within a single codebase, whereas MSA segments functionalities into independent microservices. Modulith introduces distinct internal packages for subdomains within a monolithic structure, offering a balance between modularity and coherence.

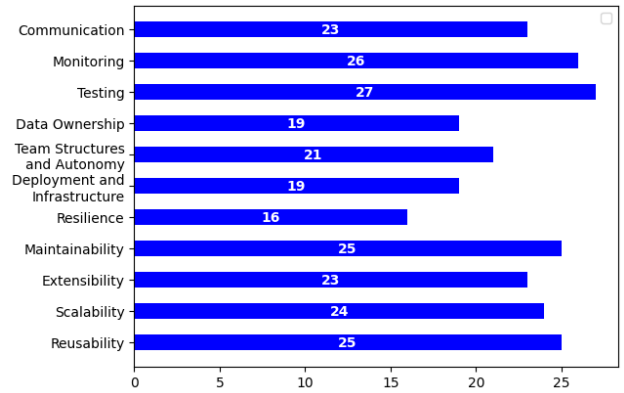

Figure 5. Decomposition factors

During the interviews, the Modulith concept surfaced as a notable and strategic adoption approach. Among the practitioners engaged, a total of 13 individuals highlighted their experiences with the Modulith strategy, elucidating that they have successfully implemented this approach in at least one scenario. Remarkably, 8 of these practitioners resoundingly advocated that the Modulith serves as an exceptional entry point for contemporary applications. This strategic choice affords them the dual advantages of the streamlined monolithic architecture and the adaptive prowess inherent in MSA.

7. Discussion

This section begins with a comprehensive summary addressing both RQ1 and RQ2. Following the summary, the section elaborates on the implications drawn from the findings section, encompassing insights relevant to both practitioners and researchers.

A. Summary to answer RQ1 and RQ2

In this study, we delved into the intricate landscape of MSA adoption by practitioners, seeking to decipher the factors guiding their decision-making processes and unveil the strategies they employ during the adoption journey. The insights gathered from various practitioners across diverse domains are presented as an overview map in Figure 6 , these insights offer valuable perspectives on the intricacies of adopting MSA.

1) Factors influencing decision-making (RQ1)

RQ1 aims to uncover the intricate processes guiding practitioners' decisions as they adopt MSA within organizational contexts. This investigation spans two essential dimensions: "Time to use or the reason to use" and "Decision-making factors or what factor triggers the use of MSA"

each contributing unique insights into the dynamics of MSA implementation.

The decision of the right time to start adopting MSA is shaped by various influences that guide practitioners' decisions on when to adopt this architectural paradigm. These influences encompass a spectrum of factors, including early adoption driven by the pursuit of enhanced scalability and agility, as well as reactive adoption prompted by challenges posed by monolithic systems. Strategic long-term planning plays a role in aligning MSA adoption with an organization's overarching goals, while post-market release and evolving customer demands can catalyze the shift towards a more flexible architecture. The modernization of legacy systems for improved maintainability and scalability can also trigger the decision to embrace MSA. Integrating updated technology stacks and adopting cloud services can act as enablers, fostering an environment conducive to MSA implementation. Moreover, the impact of market dynamics and industry trends further influences the timing of MSA adoption. In essence, the right time to use reflects a dynamic interplay of factors that underscores the nuanced nature of practitioners' decisions in adopting MSA.

8 motivating factors start the course of MSA integration. Reusability emphasizes the value of modular components reusable across projects, enhancing consistency and efficiency. Scalability emerges as a key driver, urging MSA adoption to tackle varying workloads and accommodate growing needs. Extensibility underscores the seamless integration of new functionalities into existing services, vital for accommodating evolving user requirements. Maintainability surfaces as a cornerstone, emphasizing streamlined management and upkeep of microservices. Acknowledging Technology Heterogeneity, organizations opt for adaptable frameworks like MSA to accommodate diverse technology stacks. Faster Development Cycles leverage MSA's modularity, enabling agile methodologies and swift software delivery. The inclusion of DevOps practices signifies a unified approach, accelerating deployment and enhancing collaboration. Lastly, Resilience underscores MSA's capacity for fault tolerance and robustness, ensuring reliability.

The integration of DevOps principles significantly impacts the adoption of MSA projects, playing a transformative role in streamlining processes and fostering a culture of collaboration and efficiency. The key principles of DevOps include CI/CD, automation, collaboration, and monitoring. By embracing these principles, organizations can ensure a seamless and iterative development process, allowing for quicker releases and enhanced software delivery. In the context of MSA adoption, the principles of DevOps align closely with the core tenets of microservices, promoting agility, scalability, and faster development cycles. The automation and continuous integration fostered by DevOps practices complement the modularity and independence inherent in microservices. This integration results in a holistic approach, where the collaborative and automated nature of DevOps amplifies the advantages offered by MSA, ultimately contributing to more effective and responsive organizational software development.

Every practitioner's adoption of MSA is grounded in a specific rationale, even among the 2 practitioners influenced by Market and Industry Trends. Initially, they employ MSA as a part of their marketing strategy. Hence, the decision of using is always based on at least one factor such as reusability, scalability... The overarching conclusion derived from the findings of RQ1 underscores that **"MSA should be approached as a solution rather than a primary architectural choice for organizational projects, except in cases where a significant degree of reusability is evident"**.

2) Key practices and strategies (R2)

RQ2 delves into the realm of key practices and strategies employed by practitioners during the adoption of MSA in their projects. This exploration encompasses a twofold perspective: Adoption Strategy and Decomposition Factors. These dimensions shed light on the pragmatic approaches' practitioners undertake and the decisive considerations they weigh when embarking on the journey of MSA integration.

Adoption strategy: Two adoption strategies were identified, with practitioners selectively deploying each strategy to align with their specific contexts. The Strangler strategy emerged as a preferred approach, allowing a gradual transition from monolithic architecture to MSA by replacing components incrementally. The Demolition strategy also gained traction, enabling practitioners to dismantle the monolithic application entirely and reconstruct it as a microservices-based system.

Within the Strangler strategy, we discovered practitioners employing various decomposition methodologies to guide the transition from monolithic architecture to MSA. These included DDD, EDD, Functional Decomposition, and Data-Driven Decomposition. Similarly, the Demolition Strategy, another prevalent approach, was associated with the use of DDD and EDD. This finding highlights the versatility of decomposition methodologies in guiding practitioners through the intricate process of transitioning to microservices, thus contributing to the broader understanding of effective MSA adoption strategies.

Decomposition factors: The factors that are considered in the process of decomposition are critical cornerstones in shaping the landscape of MSA adoption. Among these decisive factors, Reusability emerges as a guiding principle, advocating for modular components that can be efficiently reused across projects, streamlining development efforts. Scalability, a resounding concern, stands as a driving force behind MSA adoption, ensuring the architecture's capacity to handle evolving workloads and future growth demands. Extensibility, a central consideration, underscores the importance of seamlessly integrating new functionalities into the architecture, enabling adaptability to changing business

needs. Maintainability takes the forefront as a linchpin for sustainable development, emphasizing the organized management and upkeep of microservices. The focus on Resilience highlights the architecture's capacity to ensure fault tolerance and robustness, contributing to a dependable system. Deployment and Infrastructure factors are pivotal, reflecting the adoption of modern practices like containerization and orchestration to enhance deployment and management efficiency. Team Structures and Autonomy considerations champion the establishment of dedicated teams for microservices, fostering autonomy and expediting development cycles. Data Ownership, a crucial aspect, stresses the clear delineation of data ownership to avoid conflicts during decomposition. The Testing factor emerges as a beacon for quality assurance, addressing the intricate functionality and reliability of individual microservices. Monitoring, an integral consideration, empowers practitioners to proactively understand and address performance issues within the microservices ecosystem. Communication, a key element, underscores the need for effective collaboration between microservices to maintain coherence and harmonization within the distributed system. In summation, these decomposition factors collectively sculpt the framework of MSA adoption, their intricate interplay shaping an architecture that aligns with organizational objectives, resilience, and responsiveness.

Testing: Testing assumes a central role in the adoption of MSA, as highlighted by insights from 27 out of 30 practitioners in our research. This unanimous acknowledgment underscores the critical importance of testing in ensuring the reliability, scalability, and maintainability of microservices and the overall system. Rigorous testing scenarios, encompassing unit tests, integration tests, and end-to-end tests, are imperative to identify and rectify issues related to service communication, ensuring the robustness of each microservice. Scalability validation through performance, load, and stress testing is essential to ascertain that microservices can scale independently to meet varying demands. Testing procedures, such as contract testing, contribute to maintaining consistency across modules, ensuring that changes to one microservice do not adversely impact others. Additionally, testing strategies that simulate challenges in distributed systems address issues related to network latency, data consistency, and error handling. Embracing a DevOps culture, continuous testing becomes integral to the CI/CD pipeline, aligning with principles of agility and responsiveness. This comprehensive testing approach minimizes runtime errors in production, fostering the success of MSA projects in real-world organizational contexts.

Modulith: The concept of a Modulith emerged as a pragmatic bridge between monoliths and microservices. Practitioners favored the Modulith approach, capitalizing on the simplicity of a monolithic structure while retaining the modularity of microservices. It offered an intermediary phase wherein the system is decomposed into microservices but remains implemented as a monolith, providing flexibility to transition modules based on scalability triggers.

In summary, following the decision to embrace MSA, the subsequent adoption process assumes paramount importance. The consensus among most practitioners underscores the significance of a gradual, staged adoption approach. Moreover, it's noteworthy that the decomposition process isn't merely a methodology; it encompasses a cluster of indispensable factors, including reusability, scalability, and others. Nevertheless, it's important to acknowledge that practical success in this endeavor often hinges on the practitioner's experience and expertise.

B. Implications to practitioners

1) Primary VS solution

The insights derived from these findings hold significant implications for practitioners embarking on the journey of adopting MSA. Importantly, the study emphasizes that the adoption of MSA is primarily driven by the need to overcome specific challenges rather than a one-size-fits-all architectural approach. This signifies that MSA is predominantly employed as a solution to the issues inherently present in monolithic systems, aligning with the principle of selecting the most appropriate tool for the task at hand. It's recommended that practitioners approach MSA adoption with a strategic perspective, thoroughly evaluating the organizational context and the unique challenges that necessitate such a transition. Furthermore, the findings shed light on a unique scenario where organizations aiming for a high level of reusability can contemplate adopting MSA as their core architectural approach. This is rooted in the advantageous quality of reusability that MSA offers. Notably, certain microservices, like Authentication and Payment microservices, can be viewed as pre-designed components that can be seamlessly integrated across a range of projects, even spanning different business domains. As a result, practitioners looking to capitalize on MSA's inherent reusability strength are encouraged to consider adopting it as a foundational architecture. In this approach, these pre-designed microservices can serve as robust building blocks for various projects.

It's advised to approach MSA adoption from a dual perspective: as a tailored solution for addressing the specific challenges posed by monolithic systems and as a strategy for achieving enhanced reusability. By aligning MSA thoughtfully with contextual requirements and harnessing its advantages in terms of reusability, organizations can unlock the complete potential of this architectural paradigm. This, in turn, can drive innovation, efficiency, and competitiveness in their projects.

2) Modulith

Practitioners widely acknowledge the Modulith strategy as an effective approach for MSA. Serving as a bridge between monolithic simplicity and MSA flexibility, Modulith allows organizations to balance the benefits of both paradigms. It provides a practical stepping stone for a

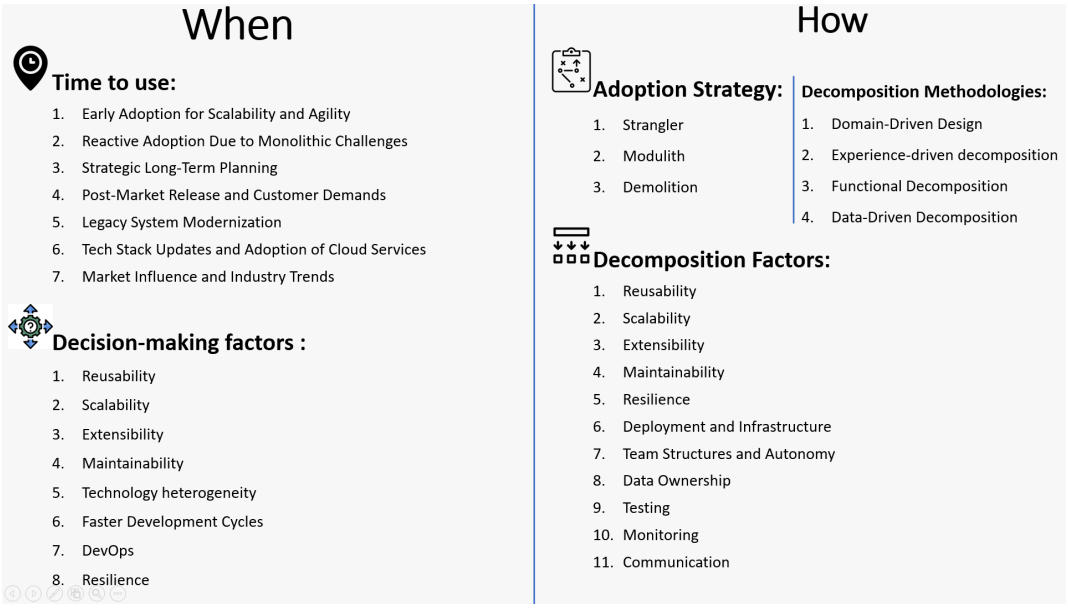

Figure 6. The integrated overview map of "When" (RQ1) and "How" (RQ2) the adoption of MSA

gradual transition, especially when an immediate shift to microservices is impractical. The strategy enables organizations to decompose monolithic applications into microservices while retaining implementation simplicity. Practitioners can identify and transition modules based on triggers like scalability requirements, minimizing disruptions and allowing flexible architecture adjustments over time. The Modulith approach leverages existing monolithic expertise, reducing the learning curve associated with full-fledged MSA transitions, ensuring a smoother shift and minimizing challenges.

C. Implications to researchers

1) Experience-based adoption

The study emphasizes a crucial need for researchers to address the influence of practitioners' experience on MSA implementation, highlighting a lack of standardized guidelines. While existing strategies like DDD provide frameworks for transitioning from monolithic systems, the absence of definitive guidelines leads to varied outcomes. This inconsistency calls for comprehensive research to develop structured and standardized methodologies. Researchers can explore factors shaping practitioners' approaches, identify challenges, and establish best practices. Clear, step-by-step frameworks considering system complexity, business domain, and scalability requirements can guide practitioners, ensuring smoother migrations, reducing uncertainties, and promoting consistent outcomes. Standardizing migration and decomposition strategies not only facilitates industry-wide MSA adoption but also enhances knowledge sharing, collaboration, and the evolution of proven methodologies.

2) Modulith approach

This study highlights the Modulith approach as an underexplored yet potentially advantageous strategy in MSA adoption. While practitioners acknowledge its promise, a lack of documented research surrounds its implementation. The unique proposition of combining monolithic simplicity with microservices' modularity presents an intriguing avenue for in-depth investigation. Researchers have a valuable opportunity to conduct comprehensive studies covering Modulith's conceptual framework, practical implementation, benefits, challenges, and comparative analyses with other MSA adoption strategies. Thorough documentation of practitioners' experiences can contribute empirical evidence on Modulith's effectiveness, adaptability, and real-world implications. Additionally, exploring its potential applications and limitations across diverse industry domains can provide valuable insights into scenarios where Modulith excels, potential challenges, and strategies to address drawbacks.

8. Threats to validity

In this section, we discuss the potential threats to the validity of our study, including construct validity, external validity, and reliability.

Construct validity: One potential threat to construct validity is the subjective interpretation of interview responses. Despite efforts to ensure clarity and consistency in our interview questions, there is a possibility of misinterpretation by both interviewees and interviewers. To mitigate this threat, we conducted pilot interviews to refine our interview protocol and ensure that questions were understood as intended. Additionally, the use of qualitative data analysis software helps enhance the accuracy of our thematic coding, minimizing the risk of misrepresenting participants' viewpoints.

External validity: As our study focuses on a specific geographical region and a selected group of practitioners,

there is a potential limitation in terms of external validity. The characteristics of the practitioners, organizations, and industries in our sample might not fully represent all potential contexts where MSA is adopted. However, we aimed to enhance external validity by selecting participants from a diverse range of domains and company sizes. While our findings may not be fully generalizable, they contribute valuable insights that can be informative for practitioners and researchers across various contexts.

Reliability: To address this potential threat, we implemented several measures to minimize observer bias and interpretation bias. Firstly, we utilized thematic coding in conjunction with an inter-rater reliability assessment to enhance the consistency of our data analysis. This involved the use of qualitative data analysis software, NVivo, to facilitate the thematic coding process. Additionally, we performed cross-analysis of responses from different interviewers, identifying any discrepancies in the extracted data. In cases of disagreement, a collaborative discussion was initiated among researchers until a consensus was reached. These steps collectively contribute to bolstering the reliability of our study's findings.

9. Conclusion

This study delved into the nuanced landscape of MSA adoption within organizational contexts. Through thorough interviews and qualitative analysis, we unveiled the intricate decision-making processes and key practices practitioners employ during MSA adoption. The research highlighted that MSA is not a one-size-fits-all solution but a deliberate choice aligned with specific organizational needs. Triggered by factors like reusability and scalability, MSA addresses specific challenges within a monolithic system. The Modulith strategy emerged as an attractive option, combining monolith simplicity with microservices' modularity for a phased transition. Future research avenues include a deeper exploration of Modulith, standardization of migration strategies, and ongoing investigation into emerging adoption challenges. Expanding the study across industries and geographies promises a more comprehensive understanding of MSA adoption practices, laying a foundation for continued research to enhance practicality and effectiveness.

References

[1] S. Newman, *Building Microservices: Designing Fine-Grained Systems*, 2 2015, iSBN: 1491950358, URL: https://www.oreilly.com/library/view/building-microservices/9781491950340.

[2] M. Ait Said, A. Ezzati, and S. Arezki, "Microservice-Specific language, a step to the Low-Code platforms," pp. 817–828, 1 2023. [Online]. Available: https://doi.org/10.1007/978-3-031-26384-2_72

[3] J. Lewis and M. Fowler, "Microservices," 2014. [Online]. Available: https://martinfowler.com/articles/microservices.html

[4] S. Newman, *Monolith to microservices*. O'Reilly Media, 9 2019, iSBN: 9781492047841, URL: https://www.oreilly.com/library/view/monolith-to-microservices/9781492047834.

[5] W. Hasselbring and G. Steinacker, "Microservice Architectures for Scalability, Agility and Reliability in E-Commerce," *IEEE International Conference on Software Architecture Workshops (ICSAW)*, 4 2017. [Online]. Available: https://doi.org/10.1109/icsaw.2017.11

[6] S. S. De Toledo, A. Martini, and D. I. K. Sjøberg, "Improving agility by managing shared libraries in microservices," pp. 195–202, 1 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-58858-8_20

[7] G. J. Blinowski, A. Ojdowska, and A. Przybyłek, "Monolithic vs. Microservice Architecture: A Performance and Scalability Evaluation," *IEEE Access*, vol. 10, pp. 20 357–20 374, 1 2022. [Online]. Available: https://doi.org/10.1109/access.2022.3152803

[8] A. Razzaq and S. A. K. Ghayyur, "A systematic mapping study: The new age of software architecture from monolithic to microservice architecture—awareness and challenges," *Computer Applications in Engineering Education*, vol. 31, no. 2, pp. 421–451, 11 2022. [Online]. Available: https://doi.org/10.1002/cae.22586

[9] I. K. Aksakalli, T. Çelik, A. B. Can, and B. Tekinerdoğan, "Deployment and communication patterns in microservice architectures: A systematic literature review," *Journal of Systems and Software*, vol. 180, p. 111014, 10 2021. [Online]. Available: https://doi.org/10.1016/j.jss.2021.111014

[10] M. Waseem, P. Liang, and M. Shahin, "A Systematic Mapping study on Microservices Architecture in DevOps," *Journal of Systems and Software*, vol. 170, p. 110798, 12 2020. [Online]. Available: https://doi.org/10.1016/j.jss.2020.110798

[11] M. Shahin, A. R. Nasab, and M. A. Babar, "A qualitative study of architectural design issues in DevOps," *Journal of Software: Evolution and Process*, vol. 35, no. 5, 9 2021. [Online]. Available: https://doi.org/10.1002/smr.2379

[12] M. Ait Said, A. Ezzati, and S. Arezki, "Microservices, a Step from the Low-Code to the No-Code," pp. 779–788, 11 2022. [Online]. Available: https://doi.org/10.1007/978-3-031-20601-6_64

[13] J. Doležal and A. Buchalcevová, "Migration from monolithic to microservice architecture : Research of Impacts on Agility," *IDIMT-2022 : digitalization of society, business and management in a pandemic : 30th Interdisciplinary Information Management Talks*, pp. 401–, 2022. [Online]. Available: https://doi.org/10.35011/IDIMT-2022-401

[14] S. Salii, J. Ajdari, and X. Zenuni, "Migrating to a microservice architecture: benefits and challenges," *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*, 5 2023. [Online]. Available: https://doi.org/10.23919/mipro57284.2023.10159894

[15] G. Lv, B. Huang, Z. Liang, M. Qin, H. Zou, and Z. Li, "Microservices: architecture, container, and challenges," *IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 12 2020. [Online]. Available: https://doi.org/10.1109/qrs-c51114.2020.00107

[16] V. Velepucha and P. Flores, "A survey on Microservices Architecture: Principles, Patterns and migration challenges," *IEEE Access*, vol. 11, pp. 88 339–88 358, 1 2023. [Online]. Available: https://doi.org/10.1109/access.2023.3305687

[17] V. Velepucha and F. Pamela, "A survey on Microservices Architecture: Principles, Patterns and migration challenges," *IEEE*

Access, vol. 11, pp. 88 339–88 358, 1 2023. [Online]. Available: https://doi.org/10.1109/access.2023.3305687

[18] R. M. Munaf, J. Ahmed, F. Khakwani, and T. Rana, "Microservices Architecture: Challenges and Proposed Conceptual Design," *International Conference on Communication Technologies (ComTech)*, 3 2019. [Online]. Available: https://doi.org/10.1109/comtech.2019.8737831

[19] Y. Wang, H. Kadiyala, and J. Rubin, "Promises and challenges of microservices: an exploratory study," *Empirical Software Engineering*, vol. 26, no. 4, 5 2021. [Online]. Available: https://doi.org/10.1007/s10664-020-09910-y

[20] M. Kolny, "Scaling up the Prime Video audio/video monitoring service and reducing costs by 90

[21] P. Di Francesco, P. Lago, and I. Malavolta, "Architecting with microservices: A systematic mapping study," *Journal of Systems and Software*, vol. 150, pp. 77–97, 4 2019. [Online]. Available: https://doi.org/10.1016/j.jss.2019.01.001

[22] S. Hassan, R. Bahsoon, and R. Kazman, "Microservice transition and its granularity problem: A systematic mapping study," *Software: Practice and Experience*, vol. 50, no. 9, pp. 1651–1681, 6 2020. [Online]. Available: https://doi.org/10.1002/spe.2869

[23] V. Bushong, A. S. Abdelfattah, A. A. Maruf, D. Das, A. Lehman, E. Jaroszewski, M. Coffey, T. Černý, K. Frajták, P. Tisnovsky, and M. Bureš, "On Microservice Analysis and Architecture Evolution: A Systematic Mapping Study," *Applied sciences*, vol. 11, no. 17, p. 7856, 8 2021. [Online]. Available: https://doi.org/10.3390/app11177856

[24] N. Alshuqayran, N. Alí, and R. Evans, "A Systematic Mapping Study in Microservice Architecture," *IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, 11 2016. [Online]. Available: https://doi.org/10.1109/soca.2016.15

[25] J. A. Valdivia, A. Lora-González, X. Limón, K. Cortés-Verdín, and J. O. Ocharán-Hernández, "Patterns Related to Microservice Architecture: a Multivocal Literature Review," *Programming and Computer Software*, vol. 46, no. 8, pp. 594–608, 12 2020. [Online]. Available: https://doi.org/10.1134/s0361768820080253

[26] D. N. Elisabetta, "Design patterns and anti-patterns in microservices architecture : a classification proposal and study on open source projects," 5 2022. [Online]. Available: https://www.politesi.polimi.it/handle/10589/186745

[27] A. Akbulut and H. G. Perros, "Performance analysis of microservice design patterns," *IEEE Internet Computing*, vol. 23, no. 6, pp. 19–27, 11 2019. [Online]. Available: https://doi.org/10.1109/mic.2019.2951094

[28] S. Li, H. Zhang, Z. Jia, Z. Li, C. Zhang, J. Li, Q. Gao, J. Ge, and Z. Shan, "A dataflow-driven approach to identifying microservices from monolithic applications," *Journal of Systems and Software*, vol. 157, p. 110380, 11 2019. [Online]. Available: https://doi.org/10.1016/j.jss.2019.07.008

[29] Z. Li, C. Shang, J. Wu, and Y. Li, "Microservice extraction based on knowledge graph from monolithic applications," *Information Software Technology*, vol. 150, p. 106992, 10 2022. [Online]. Available: https://doi.org/10.1016/j.infsof.2022.106992

[30] M. Söylemez, B. Tekinerdoğan, and A. K. K. Tarhan, "Challenges and Solution Directions of Microservice Architectures: A Systematic Literature review," *Applied sciences*, vol. 12, no. 11, p. 5507, 5 2022. [Online]. Available: https://doi.org/10.3390/app12115507

[31] F. Auer, V. Lenarduzzi, M. Felderer, and D. Taibi, "From monolithic systems to Microservices: An assessment framework," *Information Software Technology*, vol. 137, p. 106600, 9 2021. [Online]. Available: https://doi.org/10.1016/j.infsof.2021.106600

[32] R. M. Munaf, J. Ahmed, F. Khakwani, and T. Rana, "Microservices Architecture: Challenges and Proposed Conceptual Design," *International Conference on Communication Technologies (ComTech)*, 3 2019. [Online]. Available: https://doi.org/10.1109/comtech.2019.8737831

[33] X. Zhou, S. Li, L. Cao, H. Zhang, Z. Jia, C. Zhong, Z. Shan, and M. A. Babar, "Revisiting the practices and pains of microservice architecture in reality: An industrial inquiry," *Journal of Systems and Software*, vol. 195, p. 111521, 1 2023. [Online]. Available: https://doi.org/10.1016/j.jss.2022.111521

[34] S. Brinkmann and S. Kvale, *Doing interviews*, 1 2018, iSBN: 9781529716665, DOI: https://doi.org/10.4135/9781529716665, URL: https://www.sagepub.com/hi/nam/doing-interviews/book259212.

[35] E. Welsh, "Dealing with Data: Using NVivo in the Qualitative Data Analysis Process," *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, vol. 3, no. 2, 5 2002. [Online]. Available: https://www.mendeley.com/catalogue/bab4fb29-4c2a-31a2-bf74-52f708ea0cbc/

[36] M. Q. Patton, "Qualitative evaluation and research methods, 2nd ed." *Sage Publications, Inc*, 1 1990. [Online]. Available: https://psycnet.apa.org/record/1990-97369-000

[37] H. Chawla and H. Kathuria, "Evolution of microservices architecture," pp. 1–20, 1 2019. [Online]. Available: https://doi.org/10.1007/978-1-4842-4828-7_1

[38] L. Cao and C. Zhang, "Implementation of Domain-oriented Microservices Decomposition based on Node-attributed Network," *11th International Conference on Software and Computer Applications*, 2 2022. [Online]. Available: https://doi.org/10.1145/3524304.3524325

[39] R. A. Schmidt and M. Thiry, "Microservices identification strategies : A review focused on Model-Driven Engineering and Domain Driven Design approaches," *15th Iberian Conference on Information Systems and Technologies (CISTI)*, 6 2020. [Online]. Available: https://doi.org/10.23919/cisti49556.2020.9141150

[40] S. Tyszberowicz, R. Heinrich, B. Liu, and Z. Liu, "Identifying microservices using functional decomposition," pp. 50–65, 1 2018. [Online]. Available: https://doi.org/10.1007/978-3-319-99933-3_4

[41] R. Chen, S. Li, and Z. Li, "From Monolith to Microservices: A Dataflow-Driven Approach," *24th Asia-Pacific Software Engineering Conference (APSEC)*, 12 2017. [Online]. Available: https://doi.org/10.1109/apsec.2017.53

AIT SAID Mehdi A Senior Software Architect and Ph.D. student in Computer Science with a focus on Software Engineering from the Faculty of Science and Techniques of Settat, Morocco, he brings a track record of successful project deliveries, demonstrating his commitment to crafting scalable and innovative software solutions. His passion lies in optimizing development processes and leveraging the latest technologies to deliver high-quality products. With a deep understanding of software architecture, he is dedicated to advancing the field and contributing to the ever-evolving field of technology.

EZZATI Abdellah A professor in computer sciences at the Faculty of Science and Techniques, Settat, Morocco, is an expert in Computer Architecture, Networks, and Security. His proficiency spans Wireless Networks, Cloud Computing, and Network Security. With a focus on both theory and practical application, he has contributed significantly to areas such as Computer Networks Security, Network Management, and Information Security

MIHI Soukaina A PhD in computer science and artificial intelligence from the faculty of science and techniques of Settat, Morocco. She is engineer in computer science from the national school of applied sciences of Marrakesh, Morocco and has a Master degree in Artificial Intelligence from the national institute of applied sciences of Lyon, France. She has several published papers in Scopus and web of science journals. Her fields of interest include natural language processing, deep learning, sentiment analysis, information systems and health systems

BELOUADDANE Lahcen A Ph.D. student in Computer Science and IT Infrastructure at the Faculty of Science and Techniques in Settat, Morocco, is dedicated to advancing knowledge in the dynamic intersection of computing and technology infrastructure. With a strong academic background, he is actively engaged in cutting-edge research, exploring innovative solutions to challenges in the realm of computer science.