

Diana Geibel

DevOps:
A Supportive Implementation Decision Framework

Masterarbeit

Themensteller: Michael Hüttermann

Vorgelegt in der Masterprüfung PO 2015
im Studiengang Information Systems M. Sc.

der Wirtschafts- und Sozialwissenschaftlichen Fakultät der Universität zu Köln

Köln, 23. Februar 2023

Contents

Tables	iv
Figures.....	v
List of Abbreviations.....	vi
1 Introduction.....	1
2 Theoretical Background.....	3
2.1 Information Systems Development.....	4
2.1.1 Traditional Software Development	4
2.1.2 Agile Approaches.....	6
2.1.3 Characterizing DevOps	8
2.1.4 Modern Information Systems Development	12
2.2 Systematic Literature Review Approaches	14
2.2.1 Systematic Literature Review by Webster & Watson (2002).....	14
2.2.2 Systematic Literature Review by Levy & Ellis (2006).....	15
2.3 Frameworks.....	16
3 Methodology: Systematic Literature Review	18
3.1 First Stage: Data Input	18
3.2 Second Stage: Publications Selection	20
3.3 Third Stage: Systematic Literature Review Outputs.....	22
4 Results.....	23
4.1 Results I: Current Body of Knowledge on DevOps.....	23
4.1.1: DevOps Practices	28
4.1.2: DevOps Principles	29
4.1.3: DevOps Groups.....	30
4.1.4: DevOps Management.....	31
4.1.5: DevOps Tools	32
4.2 Results II: DevOps Definitions	33
4.2.1: DevOps Practices	36
4.2.2: DevOps Principles	36
4.2.3: DevOps Groups.....	37
4.2.4: DevOps Management.....	37
4.2.5: DevOps Tools	37
4.3 Results III: Supportive DevOps Implementation Decision Framework	37

4.3.1	Challenges and Goals of IS Development and DevOps Implementation .	38
4.3.2	DevOps Definitions and Target Groups.....	41
5	Discussion.....	43
5.1	Results I: Current Body of Knowledge on DevOps.....	43
5.1.1:	DevOps Practices	44
5.1.2:	DevOps Principles	45
5.1.3:	DevOps Groups.....	47
5.1.4:	DevOps Management.....	47
5.1.5:	DevOps Tools	48
5.1.6:	Merging DevOps Categories.....	49
5.1.7:	Practice-oriented DevOps research.....	50
5.2	Results II: DevOps Definitions	51
5.3	Results III: A Supportive DevOps Implementation Decision Framework	54
5.3	Further general Implications and Limitations	56
6	Conclusion	57
	Literature	iv
A.	Appendix: Journal List and Proceedings Publication (Levis & Ellis, 2006)	x
B.	Appendix: Overview of all Databases.....	xi
C.	Appendix: Timely Overview of Publications	xiv
D.	Appendix: DevOps Definitions Overview	xiv
	Eidesstaatliche Versicherung	xxi

Tables

Table 1: Challenges and goals of traditional IS Development.....	6
Table 2: Challenges and goals in agile software development	7
Table 3: Challenges and goals of DevOps	11
Table 4: Task areas in modern IS Development	12
Table 5: Challenges and goals in modern IS Development	13
Table 6: Methodology Stages	18
Table 7: List of used databases	19
Table 8: Intermediate Results of the Systematic Literature Review.....	21
Table 9: Final list of all considered publications	22
Table 10: Concept Matrix	24
Table 11: Common DevOps Definitions within DevOps research.....	34
Table 12: DevOps Definitions	35

Figures

Figure 1: Concept Matrix (Webster & Watson, 2002, p. xvii)	14
Figure 2: The three stages of effective literature (Levy & Ellis, 2006, p. 182)	15
Figure 3: Total of referenced DevOps categories within the systematic literature review	26
Figure 4: Research Methods of all considered publications	27
Figure 5: Publication types of all considered publications	27
Figure 6: Supportive DevOps Implementation Decision Framework.....	39

List of Abbreviations

IS	Information Systems
----	---------------------

1 Introduction

Agile working has been transformed by DevOps (Amaro et al., 2023, p. 1; Dornenburg, 2018, p.71). The current world of information systems (IS) development is characterized by continuous new requirements, necessity for high flexibility and responsiveness, agile approaches, and fast pace (Alt et al., 2017, p. 21; Highsmith, 2010, pp. 166-169). IS development is still a young discipline that is required to continuously develop and improve (Ouhbi & Pombo, 2020, p. 1). In traditional IS development and many agile approaches, development and operations are kept separate. Thereby, the two areas work separately from each other and under individual objectives (Hüttermann, 2012, pp. 15-16; Wolff, 2016, pp. 1-5). Within associated processes, the responsibility for a software product is terminated by the developers at the point of the software release. Then, operations become responsible for further actions (Alt et al., 2017, pp. 23-25). This process causes many problems such as releases with errors or delays, conflicts in collaboration, not fulfilled requirements, and negative user experience (Alt et al., 2017, pp. 23-25; Disterer, 2011, pp. 48-50).

These issues called for change in IS development approaches and required improved collaboration. An approach to counteract these circumstances is presented as *DevOps*. DevOps is a combination of the two terms “Development” and “Operations”. Generally, DevOps represents a collaborative view of development and operation within IS development. Thereby, DevOps supports the integration of operations into the process of IS development (Beetz & Harrer, 2022, p. 70; Beyer et al., 2018, p. 7). The use of the DevOps approach in organizations is intended to accelerate the software development process and improve the quality of the software (Erich et al., 2017, p. 1). Just like overarching IS development, DevOps is impacted by its continuously changing environment. Thus, DevOps is still an evolving topic underlying constant change and adaptations within practice and research (Beyer et al., 2018, p. 7). However, it “is becoming a main competency required by the software industry” (Perez et al., 2022, p. 1).

In scientific research, the topic of DevOps has already been considered from several viewpoints. For about 15 years now, the topic has received attention in the scientific literature and especially in information systems research. However, the term DevOps is generally only defined in the context of its use and only a few approaches

exist for a uniform definition (Jabbari et al., 2016, p. 1; Wiedemann et al., 2019, p. 48). Based on this, a variety of DevOps definitions already exist in several contexts (for example, Bass et al., 2015; Dyck et al., 2015; Humble & Farley, 2010; Humble & Molesky, 2011; Jabbari et al., 2016; Hüttermann, 2021). Additionally, previous literature on DevOps has focused primarily on practices, benefits, and culture within this topic area (for example, Akbar et al., 2022; Díaz et al., 2021; Faustino et al., 2022; Leite et al., 2020).

Based on current DevOps research, this thesis will take a differentiated perspective on the topic of DevOps. In particular, the definition problem is to be counteracted and the conceptualization of DevOps is to be promoted. Thereby, DevOps implementation will be considered regarding respective goals and given challenges in IS development. In this regard, this thesis will elaborate solutions to answer the underlying research question:

RQ: How can an implementation of DevOps help to achieve respective goals or to tackle given challenges in information systems development?

To answer the underlying research question, the thesis follows a systematic literature review approach to generate three interconnected results. First, the systematic literature review approach will be conducted. This approach will follow guidelines provided by Webster & Watson (2002) and Levy & Ellis (2006). Thereby, a concept matrix will be created to categorize current research streams of DevOps research. Thus, the aim is to present a representation of the current body of knowledge on DevOps. Second, an overview of possible DevOps definitions as an aggregation of different lines of research, respectively domains, affected by DevOps will be provided. Therefore, possible DevOps definitions will be derived from the considered publications of the systematic literature review. Third, a supportive DevOps implementation decision framework will be derived, which provides an overview of which line of DevOps definition should be utilized to implement DevOps.

In general, the systematic literature review approach will be used to answer the underlying research question. The guidelines of Webster & Watson (2002) and Levy & Ellis (2006) form the basis for the underlying methodology. They support the selection of quality IS publications as a foundation for this thesis qualitative results (Levy & Ellis, 2006, pp. 181-182; Webster & Watson, 2002, pp. xiii-xiv). Within the

process of the systematic literature review, corresponding publications will be included based on their relevance regarding the topic area of DevOps. They will be selected for the further procedure and results elaboration. Additionally, the publications will be examined individually based on their content on referenced concepts, state-of-the-art DevOps, and DevOps definitions. Consequently, the supportive DevOps implementation decision framework is conducted based on the first two interconnected results.

Furthermore, a theoretical background to IS development will be provided. This will characterize the areas of IS development, with a special emphasis on agile and DevOps. In this context, goals and challenges in these areas will be considered. Furthermore, systematic literature review approaches will be explained as well as the functionalities of a framework. Based on the underlying methodology, the interconnected results of this thesis will be presented and subsequently discussed regarding their implications and limitations for DevOps research and practice. In conclusion, a summary will capture the main findings to answer the underlying research question. Thereby, insights and recommendations for research and practice will be provided.

2 Theoretical Background

Compared to traditional sciences, software engineering is a relatively young discipline. It emerged in the 1960s to 1970s. At that time, software features were developed within years and software engineering teams were clearly separated into development and operations (Beetz & Harrer, 2022, p. 70; Dornenburg, 2018, pp. 71-73). Quickly, the traditional engineering approaches proved unable to cope with new requirements due to the increased digitalization. Thereby, continuous changes evolved in software engineering approaches and technology environments. New attempts, such as Agile and DevOps, emerged to better respond to the demands of this changing IS landscape (Alt et al., 2017, p. 27; Dornenburg, 2018, pp. 71-73). Nowadays, several hundred releases occur on a daily basis. They are characterized by new approaches, methods, and team structures (Beetz & Harrer, 2022, p. 75; Dornenburg, 2018, pp. 71-73).

In the following, this chapter gives an overview of the developed IS environment within a retrospect. Thereby, various influencing concepts on IS

development are considered. Characteristics, challenges, and goals of these concepts will be pointed out. Furthermore, the connection between agile and DevOps is given through this retrospect and the term DevOps will be characterized in detail. Additionally, an overview of systematic literature review approaches will be given as well as an introduction into characteristics of frameworks.

2.1 Information Systems Development

Information Systems has been practiced and researched since the 1970s. The fast-moving and changing field has undergone many developments (Dornenburg, 2018, p. 71; Hemon-Hildgen et al., 2020, p. 3). In the following, the area of IS development and its transformations will be reviewed. First, traditional IS is described, and based on this, modern approaches to IS development, in particular agile and DevOps, are characterized. Throughout the entire consideration, a focus is placed on the goals and challenges in the individual sections.

2.1.1 Traditional Software Development

Traditional IS development approaches dominated the industry for more than 25 years (Sommerville, 2020b, p. 16). Organizations seeking to automate processes in their company purchased software to be customized or developed for their specific requirements. Thereby, IS development processes largely rely on implementing software within one major project. The goal is to have a first major release that includes all necessary features of the software. Thereby, the process of software development is precisely planned to generate a software as the output. Thus, this process leads the organization to become the customer who sets the requirements (Sommerville, 2020b, pp. 16-18; Sommerville, 2020a, pp. 11-14; Wiedemann et al., 2019, p. 46).

Traditional software development primarily followed waterfall methods. These were based on a bottom-up principle, in which requirements were predefined by the customers and contractually agreed upon. Then, requirements are forwarded to development teams to be implemented. After the implementation process, the software is then again transferred to the customer (Sommerville, 2020b, pp. 16-26; Sommerville, 2020a, pp. 11-14).

This process requires project management to fulfill the contract requirements. In general, requirements represent a measurement for the customers. Only by fulfilling the requirements, the software can be confirmed as successfully implemented

(Sommerville, 2020b, pp. 16-26; Sommerville, 2020a, pp. 11-14; Wiedemann et al., 2019, p. 46).

A characteristic of traditional approaches is distinct operating structures. Thereby, traditional software engineering approaches follow defined business and development processes in order to implement software successfully. Partitioners in the process have clear responsibilities and task areas (Sommerville, 2020b, pp. 16-18).

After requirements engineering, the defined requirements are forwarded to the development teams. Developers then implement the stated requirements and test the software. Afterwards, the software is delivered to the customers (Sommerville, 2020b, pp. 16-26). Simultaneously, the responsibility for the running system is transferred to IT operations, which are responsible for maintaining the software in terms of software updates and bug fixing. Development teams are no longer involved in the process and further steps (Sommerville, 2020b, pp. 16-26; Wiedemann et al., 2019, p. 46).

In traditional IS development, development and operations teams were kept separate from each other. Development teams were responsible for developing releases and operations teams for maintaining the software in operation (Leite et al., 2020, pp. 2-3; Ozkaya, 2020, pp. 3-4; Rowse & Cohen, 2021, p. 6785; Silva et al., 2018, pp. 1-2). This culture has led to a lack of communication between the development and operations teams. Among other things, operations teams were left out of requirements processes and general considerations. These circumstances cause delays in software updates and conflicts between these two teams, resulting in ineffective processes (Leite et al., 2020, pp. 2-3; Lwakatare et al., 2016b, pp. 91-92; Ozkaya, 2020, pp. 3-4; Rowse & Cohen, 2021, p. 6785; Silva et al., 2018, pp. 1-2).

Consequently, IS development industry is challenged by the traditional approaches. On the one hand, these approaches require highly preparatory work regarding requirements engineering and resource planning. On the other hand, traditional processes are usually lacking in continuous feedback cycles. Feedback cycles are necessary to verify the achievement of requirements. Additionally, traditional IS development is lacking in flexibility and responsiveness. Thus, it is challenged by changing business and legal requirements. In summary, these circumstances led to long and complex software development processes and various other problems. Moreover, traditional IS development is characterized by a large amount of failed and delayed projects (Sommerville, 2020b, pp. 16-26; Sommerville, 2020a, pp. 11-14).

Given the challenges and problems of traditional software development, as presented in *Table 1*, more flexible software development approaches arose within the industry. Current software development is characterized by agile, flexible, and iterative approaches (Sommerville, 2020b, pp. 16-18). This allows fast decision-making and more sustainable IS development for companies (Sommerville, 2020b, pp. 16-26; Wiedemann, 2018, pp. 4931-4932). In general, the perception of software and its development changed as well as the requirements that drive software development. Legal and regulatory requirements have to be implemented faster into software and are constantly changing. It summarizes into a rapidly changing environment where traditional approaches largely fail to succeed (Sommerville, 2020b, pp. 16-26).

Table 1: Challenges and goals of traditional IS Development

Traditional IS Development	
Challenges	Goals
• Changing environment and requirements • Long planning process and development • Failure of projects	• Automation of processes • Software that achieves stated requirements

Note: Adapted from *Modernes Software-Engineering* (pp. 16-26), by I. Sommerville, 2020, Pearson Deutschland (ISBN 978-3-86894-396-2); *Engineering Software Products: an Introduction to Modern Software Engineering* (pp. 11-14) by I. Sommerville, 2020; Pearson Education Limited (978-0-13-521064-2); “Research for Practice: The DevOps Phenomenon” by Wiedemann et al., 2019, *Communications of the ACM*, 62(8), 44-49 (10.1145/3331138).

2.1.2 Agile Approaches

First agile approaches arose in the 1990s as an alternative to traditional IS development. The changing environment and requirements due to increased digitalization as well as new technologies were difficult to handle with conventional approaches (Hemon et al., 2019, pp. 3-4; Hemon et al., 2020, pp. 3-7; Smart, 2018, p. 56). Therefore, agile methods were introduced to the market to increase success of software development. Nowadays, agile is the norm for software development projects. In general, agile approaches describe the continuous development and releasing of a software product which increases in functionality and usage over time (Dornenburg, 2018, p. 71; Hemon et al., 2019, pp. 3-4; Hemon-Hildgen et al., 2020,

pp. 3-7; Smart, 2018, p. 56). In addition, agile approaches support short iterative feedback and release cycles in software development (Dornenburg, 2018, p. 71; Hemon-Hildgen et al., 2020, pp. 33; Stoica et al., 2013, pp. 64-66). Feedback is an essential part of agile approaches as it is used to directly evaluate and influence the development process (Dornenburg, 2018, pp. 72-73). The aim of agile approaches is to increase release frequencies and software quality to meet the needs of the changing software market and its requirements (Hemon-Hildgen et al., 2020, p. 33; Hüttermann, 2021, p. 10; Krey et al., 2022, p. 7306). Common agile methods are for example Scrum or Extreme Programming. These approaches already succeeded to remove barriers, such as non-iterative processes, in software development (Ozkaya, 2019, p. 3).

Agile methods are applied with a focus on development teams. In general, agile approaches support collaboration, flexibility, and processes such as planning, testing, and integration of these teams (Dornenburg, 2018, p. 71; Hemon-Hildgen et al., 2020, pp. 3-7; Stoica et al., 2013, pp. 64-66). Development teams that follow agile approaches have been proven to increase their performances (Krey et al., 2022, p. 7297; Lwakatare et al., 2016a, p. 2). However, software engineering consists of development and operations processes. In contrast to development teams, operations teams improve less from agile methods and are usually neglected (Lwakatare et al., 2016a, pp. 1-2). Practiced agile methods tend to cut out the operations and deployment part of software engineering which already was practiced in traditional software engineering approaches (Leite et al., 2020, pp. 2-3). This refers to long delays and severe communication problems in software engineering environment (Hemon-Hildgen et al., 2020, pp. 3-6). As seen in practice nowadays, there is a trend towards frequent development cycles. Thereby, the difficulties of separating development and operations are addressed (Silva et al., 2018, p. 1). Agile offers an approach to improved software development, yet the field struggles with several challenges. Nevertheless, this approach pursues goals to improve software development (Beck et al., 2001; Sommerville, 2020b, pp. 34-35). A summary of these challenges and goals is shown in *Table 2*. To tackle the bottleneck of missed attention on operations in agile methods, DevOps evolved (Hemon-Hildgen et al., 2020, p. 3).

Table 2: Challenges and goals in agile software development

Agile Software Development

Challenges	Goals
• Separation of development and operations • Understanding and spreading of the agile mindset • Communication and collaboration problems	• Include customers and customer satisfaction • Accept changes and changing requirements • Incremental development and deployment • Increased release frequencies and software quality • Simplicity within software development and all processes • Focus on team collaboration and communication

Note: Adapted from *Manifesto for Agile Software Development*, by Beck et al., 2001 (<https://agilemanifesto.org/>); *Modernes Software-Engineering* (pp. 34-35), by I. Sommerville, 2020, Pearson Deutschland (ISBN 978-3-86894-396-2)

2.1.3 Characterizing DevOps

The DevOps history began in 2008 at the Agile Conference in Toronto (Ghantous & Gill, 2017, p. 2; Hemon-Hildgen et al., 2020, p. 3; Ozkaya, 2020, p. 3). There, Debois (2008) introduced the idea of DevOps. By stating needs and values of the software engineering industry, Debois challenged the barriers between development and operation in the industry (Hemon-Hildgen et al., 2020, p. 3). The term “DevOps” can be separated into two individual parts. “Dev” represents development and “Ops” represents operations within an IS development process (ibid.).

DevOps extends previous agile approaches by encouraging the relevance of operations in software development to remove existing barriers to flexibility, automation, and communication (Hemon-Hildgen et al., 2020, pp. 7-8; Ozkaya, 2020, pp. 3-5). Although Debois introduced the idea of DevOps, he did not further define the term (Hemon-Hildgen et al., 2020, p. 3). Therefore, DevOps literature and practice is still characterized by a lack of common knowledge, understanding, and evidence on DevOps (Lwakatare et al., 2016b, p. 92). For this reason, DevOps is not uniformly interpreted and referred to as an approach, philosophy, methodology, or framework (Hemon-Hildgen et al., 2020, p. 9; Lwakatare et al., 2016b, p. 92). DevOps was introduced to solve the problems of the current state of the software development industry (Hemon-Hildgen et al., 2020, pp. 3-9). However, the lack of a definition simultaneously brought solutions and new problems.

In general, DevOps research is highly influenced by practice-oriented publications which is why empirical and peer-reviewed publications are lacking (Beetz & Harrer, 2022, p. 740; Hemon-Hildgen & Rowe, 2022, pp. 568-569; Hüttermann, 2021, p. 71; Rafi et al., 2022, pp. 1-2). In addition, DevOps publications highly study benefits of DevOps implementation while a consideration on risks of DevOps implementation are limited (Beetz & Harrer, 2022, p. 70; Hemon-Hildgen & Rowe, 2022, p. 569). Furthermore, DevOps is subject to a continuous change of further evolution. Various trends have been identified in recent years that are based on the idea of DevOps. Consequently, a missing clear definition on DevOps is stated as a reason for these emerging trends (Beetz & Harrer, 2022, p. 70).

Initially, tools in the context of DevOps were often considered as the epitome of DevOps. These tools support developers and operations within the whole software development and maintaining lifecycle. In contrast to previous approaches, the focus on tools to support the operations side was new. These tools benefits were easy to grasp and multiply for organizations. Furthermore, tools allowed easy measurement of effectiveness (Ebert et al., 2016, pp. 94-98; Galup et al., 2020, pp. 48-53). Often, these tools relate to tasks in development, creation, automation, monitoring, and deployment of software. As a result, there is a variety of tools for different purposes in the DevOps context (Ebert et al., 2016, pp. 94-98).

In traditional software engineering, development and operations teams were kept separate from each other. This culture has led to a lack of communication between the teams, delays in software updates, and conflicts between these two teams, resulting in ineffective processes (Leite et al., 2020, pp. 2-3; Lwakatare et al., 2016b, p. 92; Ozkaya, 2020, p. 3; Rowse & Cohen, 2021, p. 6785; Silva et al., 2018, p. 1). DevOps represents a solution to the problems of earlier approaches of software engineering by integrating operations in equal measure and provide a holistic approach. These changes influence a company's working methods, management, culture, and processes (Alt et al., 2017, p. 27-32). To establish DevOps within a company, DevOps follows several practices and principles. Principles represent guidelines for DevOps that are followed by the implementation of DevOps in an organization. Principles in DevOps represent characteristics that support the DevOps mindset. These include, for example, automation of processes, collaboration in cooperation or a holistic view of software (Beetz & Harrer, 2022, pp. 70-75). The concept of CALMS is often referred to as a

principle of DevOps. Thereby, CALMS is characterized by the holistic consideration of culture, automation, measures, lean and sharing (Rafi et al., 2022, p. 3).

In contrast, practices that are considered in the context of DevOps are precise methods for implementing these principles. They are mainly focusing on continuous approaches in IS development. These methods include, for example, approaches such as continuous integration and deployment of software, as well as the implementation of microservices instead of monolith architectures (Beetz & Harrer, 2022, p. 71). These practices have grown from agile practices and support the IS development and also the IS operations' side. In this context, developers are accompanied throughout the entire lifecycle of a software (Beetz & Harrer, 2022, p. 71; Lwakatare et al., 2016a, pp. 1-2; Zhu et al., 2016, p. 33). This counteracts the one-sided consideration of the development perspective (Hemon-Hildgen & Rowe, 2022, pp. 569-570; Zhu et al., 2016, p. 33).

By adopting DevOps, involved parties such as organizations, management, and team members often expect only benefits in terms of speed and continuity of releases and team collaboration (Ebert et al., 2016, p. 98; Gall & Pigni, 2018, p. 1). However, within a company deciding to integrate DevOps, managers must be aware that DevOps represents a transformation of collaboration and culture. This affects whole organizations, and requires strong and leading management (Ozkaya, 2019, pp. 4-5; Wiedemann et al., 2019, pp. 48-49). The way of managing DevOps within a company can affect various areas, for example a company's success, employee satisfaction, and effectiveness of the DevOps transformation (Amaro et al., 2023, p. 4). Compared to agile, management in the DevOps context is more complex based on the higher need for coordination (Hemon-Hildgen et al., 2020, p. 45). Therefore, topics such as implementing DevOps in large companies and its orchestration are particularly relevant for managing DevOps. Within an organization, DevOps can be seen as an individual process in which a transformation of the organization towards practices and principles of DevOps takes place (Galup et al., 2020, pp. 48-53; Wiedemann et al., 2019, pp. 45-49). However, DevOps implementation has no uniform approach which can be challenging especially for organizational management (Ozkaya, 2019, p. 5; Wiedemann et al., 2019, pp. 45-49).

In particular, this transformation affects development and operations teams in their way of working. Team structures in the context of DevOps aim to merge development and operations teams to increase efficiency and decrease conflicts.

Thereby, these teams also do not follow a defined approach. In DevOps, team structures are reorganized so that agreeing and linking roles, responsibilities, capabilities, and working together can be focused. Cross-functional teams are also often typical when implementing DevOps. These often include roles, skills, and responsibilities from various areas within an organization (Hemon-Hildgen et al., 2020, p. 14-15). In summary, the goal of DevOps teams is to work in a collaborative and appreciative manner (Hemon et al., 2019, pp. 2-4).

In general, transforming towards DevOps fundamentally affects an organization and its underlying collaborations. Thereby, it is guided by various challenges and goals, as seen in *Table 3*. This complicates the successful implementation of DevOps within companies (Hemon-Hildgen et al., 2020, p. 37; Wiedemann et al., 2019, pp. 47-48). However, changing legal and customer requirements, increased competition, and influences from digitalization and globalization are reasons for implementing DevOps within an organization. The goal of implementing DevOps is faster and better development and delivery of systems in order to remain competitive on the market (Amaro et al., 2023, pp. 2-3; Beetz & Harrer, 2022, p. 71).

Table 3: Challenges and goals of DevOps

DevOps	
Challenges	Goals
• Organizational understanding of DevOps • Team collaboration effectiveness and efficiency • Building knowledge and skills on DevOps • Successful DevOps implementation • Competitiveness on the market • Improving information systems	• Flexibility in software development • Decrease conflicts between development and operations • Improve communication and collaboration • Automation of processes

Note: Adapted from “Capabilities and Practices in DevOps: A Multivocal Literature Review” by Amaro et al., 2023, *IEEE Transactions on Software Engineering*, pp. 2-3 (10.1109/TSE.2022.3166626); “The path to DevOps” by E. Dornenburg (2018), *IEEE Software*, 35(5), pp. 73-74 (<https://doi.org/10.1109/MS.2018.290110337>); “Conceptualising and defining DevOps: a review for understanding, not a framework for practitioners” by A. Hemon-Hildgen & F. Rowe, 2022; *European Journal of Information Systems*, 31(5), pp. 569-570 (<https://doi.org/10.1080/0960085X.2022.21>

00061); “The DevOps Continuum: Walking the Shadowy Bridge from Information Systems Development to Operations” by M. Hüttermann, 2021, *ECIS 2021 Research Papers*, 78, p. 4 (https://aisel.aisnet.org/ecis2021_rp/78)

2.1.4 Modern Information Systems Development

The constantly changing environment through increased digitization and its influences is affecting various areas of economy and society. Thus, information systems have to accomplish new goals and requirements. Traditional approaches of software development had to change for organizations to be compatible on the market. Based on this, software development evolved over the past 50 years to tackle the challenges of this continuously changing environment (Sommerville, 2020b, pp. 16-18; Rowse & Cohen, 2021, p. 6785).

These circumstances require partitioners to a flexible and iterative behavior. As seen in *Table 4*, modern IS development has various responsibility and task areas including management, stakeholders, product and release managers, development teams, as well as software developers, testers, and architects (Sommerville, 2020b, pp. 16-26).

Table 4: Task areas in modern IS Development

Task areas	Description
Management	Management describes management tasks and responsibilities of IS development from an organizational and broad perspective.
Stakeholders	Stakeholders correspond to customers and are responsible for the developments for requirements for a system.
Product Managers	Product managers describe roles that have responsibility for a specific software product and guarantee requirements-based software development.
Release Managers	Release Managers are responsible for software releases
Development Teams	Development teams are collectively responsible for the development of software based on the defined requirements.
Software Developers	Software developers describe the individual development team member.
Software Tester	Software testers are responsible for testing software and ensuring compliance with requirements.
Software Architects	Software architects are responsible for the design of the underlying software architecture.

Note: Adapted from *Modernes Software-Engineering* (pp. 16-26; pp. 94-96, p. 264), by I. Sommerville, 2020, Pearson Deutschland (ISBN 978-3-86894-396-2); *Engineering Software Products: an Introduction to Modern Software Engineering* (pp. 11-14; p. 94; p. 282) by I. Sommerville, 2020; Pearson Education Limited (978-0-13-521064-2)

Modern software development follows different approaches to tackle the impending challenges. New approaches follow principles of agile software development, DevOps, Scrum, Extreme Programming, cloud-based software methods, and new architectures like microservices. In addition, regulatory and legal requirements regarding safety and security issues are evolving within the process of continuous digitization. Therefore, IS development has to implement constant new requirements (Sommerville, 2020b, pp. 16-26). As shown in *Table 5*, the challenges and goals in modern IS development are elaborated from the described environment as well as challenges and goals from DevOps and agile approaches.

Table 5: Challenges and goals in modern IS Development

Modern IS Development	
Challenges	Goals
- Continuously changing environment - New (legal and regulatory) requirements - Competitiveness on the market - Conflicts between development and operations - Spreading new cultural changes in the way of working	- Flexibility in software development - Short time to market - Long lived information systems - Effectiveness and efficiency in collaboration and communication - Automation of processes - Reducing errors and defects

Note: Adapted from “Are DevOps and automation our next silver bullet?” by I. Ozkaya, 2019, *IEEE Software*, 36(4), pp. 1-5 (<https://doi.org/10.1109/MS.2019.2910943>); “Productivity Gains of DevOps Adoption in an IT Team: A Case Study” by Silva et al., 2019, *Designing Digitalization ISD2018 Proceeding*, pp. 2-4 (<http://aisel.aisnet.org/isd2014/proceedings2018/ISDevelopment/8>); *Modernes Software-Engineering* (pp. 16-26), by I. Sommerville, 2020, Pearson Deutschland (ISBN 978-3-86894-396-2)

2.2 Systematic Literature Review Approaches

For research, systematic literature reviews are of particular importance. A systematic literature review simplifies theory development, helps to refine research areas, and to indicate open research fields (Webster & Watson, 2002, pp. xiii-xiv). In this way, a systematic literature review contributes to theoretical and conceptual progress in a research area (Levy & Ellis, 2006, pp. 181-182).

2.2.1 Systematic Literature Review by Webster & Watson (2002)

Webster & Watson (2002) present a general approach for systematic literature reviews with a focus on IS research. This approach aims to synthesize literature in a concept centered way. The goal of this approach is to describe key concepts of selected underlying publications through a concept matrix. Thus, the results include theoretical explanations, empirical findings, and practical examples based on the examined publications through the process of the systematic literature review. Within their described process of a systematic literature review, the authors define a three step process (Webster & Watson, 2002, pp. xvi-xxi).

The first step describes the identification of relevant literature for the further process of the systematic literature review. Thereby, a structured approach is followed which contains the identification of relevant articles through a keyword search within recommended databases like *ProQuest* or *Web of Science*. The goal is to identify and select a variety of relevant publications. Furthermore, the authors extend the publications selection by including a forward and backward search. The forward search describes the examination of publications that cite the selected publications. The backward search describes the investigation of the literature used within the selected publications (Webster & Watson, 2002, pp. xv-xvi).

Table 2. Concept Matrix					
Articles	Concepts				
	A	B	C	D	...
1		✗	✗		✗
2	✗	✗			
...			✗	✗	

Figure 1: Concept Matrix (Webster & Watson, 2002, p. xvii)

The second step contains the structured review of all selected publications. In this process, the authors reference the individual consideration of the selected literature of the previous step. Through this consideration, relevant content and concepts on the subject area will be identified within each selected publication. As a result, a concept matrix, as seen in Figure 1, will be derived. This concept matrix will include an overview of all relevant concepts (Webster & Watson, 2002, pp. xvi-xviii).

The last step includes various aspects regarding the results evaluation. Thereby, the authors emphasize the relevance of results explanation, implications, discussion, and conclusion. In this context, the authors emphasize the importance of comprehensibility and contribution of the study. Additionally, the authors give recommendations on writing style such as using past and present tense for the research. As a result of the review process, a theory will be developed to answer underlying research questions (Webster & Watson, 2002, pp. xviii-xx).

In summary, the authors refer to the improvement of studies in the IS field. By providing a process for a systematic literature review, the authors aim to improve the quality of IS research (Webster & Watson, 2002, pp. xiii-xxi)

2.2.2 Systematic Literature Review by Levy & Ellis (2006)

Levy & Ellis (2006) systematic literature review approach is guided by the concept of Webster & Watson (2002). The authors fit the Webster & Watson (2002) approach specifically into the information system research environment and elaborated it with concrete process descriptions. Thereby, the authors specifically consider IS journals and publications as well as an IS based review approach. Within their framework, the authors refer to various quality IS journals and publications to bring a quality output of the systematic literature review. Therefore, Levy & Ellis (2006) define a three-stage literature review process as seen in Figure 2. The three stages correspond to 1. *Literature Review Inputs*, 2. *Literature Review Processing*, and 3. *Literature Review Outputs*.

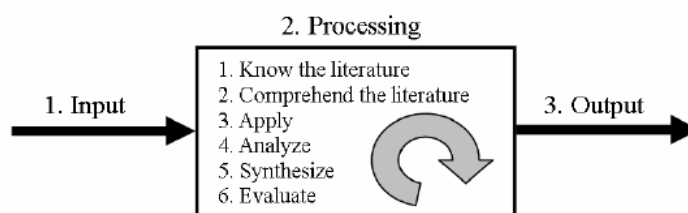


Figure 2: The three stages of effective literature (Levy & Ellis, 2006, p. 182)

In the Literature Review Input step, the authors define the basis for the systematic literature review. The aim is to provide a foundation for the selection of publications for the systematic literature review. Thereby, the authors provide a list of 50 predefined high-quality IS journals and 16 proceedings publications based on various classifications of qualitative IS journals as seen in *Appendix A*. For the selection of relevant journals, the concept of keyword search is referred to. The keyword search is used to filter relevant publications in order to create a basic set of publications for the systematic literature review. Furthermore, the authors refer to further limitations of the results of the keyword search, such as the more in-depth coverage of the examined research topic. Based on this, a set of relevant and qualitative publications should be identified. In this context, the authors also refer to the forward and backward search, which should extract further relevant publications. The aim is to find other relevant publications that are relevant to the results of the systematic literature review (Levy & Ellis, 2006, pp. 185-192).

The Literature Review Processing step refers to the creation of a concept matrix based on the approach of Webster & Watson (2002). This step identifies important concepts of the studied publications. In the concept matrix, the articles will be categorized and classified based on the identified concepts (Levy & Ellis, 2006, pp. 192-201).

Literature review outputs describe the development of a written and analytical discussion of the results of the previous steps. Thus, the authors provide assistance and approaches for the argumentative evaluation of the results of the concept matrix (Levy & Ellis, 2006, pp. 201-205).

In summary, Levy & Ellis (2006) present an effective literature review in the field of IS research and propose a framework for IS specific literature reviews. This approach aims to produce a systematic literature review adapted to the needs and requirements of IS research (Levy & Ellis, 2006, pp. 181-208).

2.3 Frameworks

In areas with lacking knowledge, understanding, or conceptualization, frameworks are usually developed to structure the underlying knowledge of the specific application area. The aim is to make knowledge applicable to the area and its environment (Macdonald et al., 2011, pp. 671-672; Okumus, 2001, pp. 327-328). In general, a framework represents a simplified and conceptualized process to increase

understanding of a identified certain area. Thereby, a framework usually maps independent and dependent variables that influence the area, an implementation process, interconnecting relationships, and a defined outcome (Leshem & Trafford, 2007, p. 98; Okumus, 2001, pp. 327-328).

In general, variables equal factors that influence the application area. In the framework, they directly or indirectly influence the modeled processes. In the context of a framework, dependent variables represent main influencing factors of the framework. Thereby, they characterize values and circumstances with direct interactions on represented processes. For example, these variables represent influencing variables such as budgets (Okumus, 2001, p. 328). In contrast, independent variables within a framework represent general influencing factors on the application area. However, they do not directly influence represented processes within a framework. These variables are influencing factors for the defined outcome (ibid.). To structure the knowledge, a framework needs to consider both independent and dependent variables that influence the implementation process and outcome (Leshem & Trafford, 2007, pp. 93-94; Okumus, 2001, pp. 327-328).

Within a framework, processes and relationships of the defined variables within the application area are also represented. Thereby, an area can be affected by several variables. These defined variables may interact with or influence other variables (Okumus, 2001, pp. 327-328). The representation of relationships and interactions is important to simplify the mapped knowledge and to make processes more understandable (Leshem & Trafford, 2007, p. 98; Okumus, 2001, pp. 327-328).

Furthermore, the outcome is the product of a developed framework. This represents the expected result based on the mapped processes and variables (Okumus, 2001, p. 328).

In summary, frameworks are used in various areas by simplifying application areas. Thereby, frameworks represent influencing factors and relationships of an underlying problem case. However, frameworks are not bound to strict representation methods and can vary in their representation styles (Leshem & Trafford, 2007, p. 98; Okumus, 2001, pp. 327-328).

3 Methodology: Systematic Literature Review

The methodology of this work is based on the systematic literature review process of Webster & Watson (2002) and Levy & Ellis (2006). The individual steps of this working methodology are described in the following. In addition, an overview of the stages is shown in *Table 6*.

Table 6: Methodology Stages

	First Stage: Data Input	Second Stage: Publications Selection	Third Stage: Systematic Literature Review Outputs
Scope	1. Selection of databases 2. Restricting the possible publications in terms of high quality	Selection of quality publications	Processing the examined publications by considering every publication.
Goal	Restricting upfront results to ensure quality of publications and final results	Provide quality and state-of-the-art publications for the systematic literature review	1. Provide a concept matrix 2. Deliver DevOps Definitions 3. Develop a supportive DevOps decision implementation framework

Note: Adapted from “A systems approach to conduct an effective literature review in support of information systems research” by Y. Levy & T. Ellis, 2006, *Informing Science*, 9., pp. 182-204; “Analyzing the past to prepare for the future: Writing a literature review” by J. Webster & R. Watson, 2002, *MIS quarterly*, pp. xiii-xxiii.

3.1 First Stage: Data Input

The first stage of the underlying methodology describes the data input for the systematic literature review. The goal of this stage is to set the publication selection for stage two of the systematic literature review. Thus, high-quality publications are selected to ensure the quality of the interconnected results (Webster & Watson, 2002, pp. xv-xvi). This stage follows the literature selection process defined by Levy & Ellis (2006). Furthermore, it will be adapted to specific circumstances of DevOps research. Based on the Levy & Ellis (2006) approach, this methodology considers quality literature review inputs in the field of IS research. To ensure the literature’s quality,

the input to the literature review in this thesis is limited by the journal list defined by Levy & Ellis (2006).

Levy & Ellis (2006) consider 50 quality IS journals for a systematic literature review within their framework that can be found in twelve defined data sources (As seen in *Appendix A*). The authors restrict the literature for a systematic literature review to only quality journals and publications based on highly ranked MIS journals. This validates the developed outputs of the thesis' systematic literature review. The journals restricted by the authors are leading peer-reviewed journals, which provide sufficient theoretical background and other sources on IS specific research topics (Levy & Ellis, 2006, pp. 181-182).

This thesis' methodology is limited to the availability and access to the data sources and recommended journals. As a result, the following systematic literature review is based on publications of the data source platforms *ProQuest database*, *EBSCOhost database*, *Elsevier (Science Direct) database*, *IEE (Comp Soc & Xplore) database*, and *ACM (Digital Lib) database*.

In addition to the recommended publications and databases, the database *Aisel AISNET* is considered. This database provides a variety of conference papers that consider the topic of DevOps (Elsevier Inc., 2023). Since the subject area of DevOps is very practice-oriented and constantly evolving (Beetz & Harrer, 2022, p. 70; Hemon-Hildgen & Rowe, 2022, pp. 568-569; Hüttermann, 2021, p. 2), these additional publications can provide further important insights into the subject area.

In conclusion, six databases are considered in the context of this thesis, as described in *Table 7*.

Table 7: List of used databases

Row Labels	Count of total considered publications
ACM	14
AISNET	47
EBSCOHOST	26
ELSEVIER	5
Forward / Backward Search	6
IEEE	41
PROQUEST	40
Grand Total	179

3.2 Second Stage: Publications Selection

This stage addresses the selection of publications based on various selection criteria defined by Webster & Watson (2002) and Levy & Ellis (2006). This ensures a qualitative content basis for the creation of the results. The aim of this stage is to select the publications for the systematic literature review. The selection of publications within the search process is described in the following. The process is based on eight steps, which apply the different selection criteria.

First, each of the six databases is searched for the keyword "DevOps" in title and/or abstract. The keyword search is used to select relevant publications and restrict the outcomes of the search (Levy & Ellis, 2006, p. 190). One goal of this thesis is to give an overview of the topic DevOps, its definitions, and application area. Based on this, the keyword "DevOps" is chosen, as this allows a broad set of results to be delivered without limiting the search results in advance. This enables to answer the underlying research question and supports the creation of intermediate results.

Second, the articles are sorted according to their language. In the following, only English-language publications are considered. On the one hand, this is based on the comprehensibility of the publications, which limits the selection. On the other hand, it is based on the possibility of the repeatability of these publications' selection.

In the third step, the search results are restricted based on the journal list of Levy & Ellis (2006), for example, only high-quality publications are considered. Regarding the case of the *Aisel AISNET* database search results, only proceedings are selected as described in stage one.

In the fourth step, all publications are filtered based on their relevance in abstract and title in relation to a contribution to this thesis. For this purpose, only publications that consider the topic of DevOps with their abstract and title are selected.

Fifth, redundant publications are removed from the list.

Sixth, all previously selected publications sorted based on the relevance of their full paper. For this purpose, only publications that consider the topic of DevOps in depth are selected for further steps. This is ensured by considering each publication individually.

In the following, an overview of removing reasons within the steps two to six of the systematic literature review stage two is shown:

1. The publication is not included in the predefined journal list.
2. The publication considers another topic than DevOps.
3. Title and Abstract: Does not consider the topic defining DevOps in depth.
4. Full Text: Does not consider the topic DevOps in depth.
5. A full text of the publication is not accessible.
6. This publication is redundant.
7. The publication is excluded due to the quality of the publication regarding a research focus.
8. The publication is written in another language.

In the next two steps, a forward and a backward search is performed as recommended by Webster & Watson (2002) and Levy & Ellis (2006) to extend the selected literature. For the forward search, all articles that cite the considered paper are checked. For the backward search, articles that are cited in the considered paper are checked. For the relevant articles, steps two to six are run again.

Based on the described selection process, *Table 8* shows an overview of the intermediate results of stage two. This results in the consideration of 13 different journals and publications, which will be considered in the further course.

Table 9 shows an overview of these. In addition, *Appendix B* shows an overview of the second stage results of each individual database.

Table 8: Intermediate Results of the Systematic Literature Review

Procedure	Description of the selection criteria	Hits
Keywords	DevOps	13.096
Language	English	11.976
Journal List	Included in the considered Journal List	162
Relevance: abstract and title	Exclude publications that are not considering DevOps	50
Full publication accessibility	Full text accessibility is necessary for further steps and results creation	44
Removing redundant publications	Publications that are duplicates will be excluded	33
Relevance: full publication	Exclude publications that are not considering the topic of DevOps in depth	26
Amount of forward search publications	All articles that cite the individually considered paper which fulfill the requirements of the previous steps	2

Amount of backward search publications	All articles that are cited in the individually considered paper which fulfill the requirements of the previous steps	5
Sum Final List		33

Table 9: Final list of all considered publications

Row Labels	Count of Publications
ACM Computing Surveys	1
AMCIS Proceedings	3
Communications of the ACM	2
ECIS Proceedings	1
European Journal of IS	3
Hawaii International Conference on System Sciences (HICSS)	5
ICIS Proceedings	1
ICSEA	1
IEEE Software	10
IEEE Trans on Software Eng	2
International Working Conference on Transfer and Diffusion of IT	1
ISD Proceedings	1
PACIS Proceedings	1
Grand Total	33

3.3 Third Stage: Systematic Literature Review Outputs

In the third stage, the individual articles are examined, and key concepts and ideas will be described. This stage has three intermediate results based on the underlying assignment and research question of this thesis.

The first intermediate result derives a concept matrix based on the approaches by Webster & Watson (2002) and Levy & Ellis (2006). This concept matrix is created based on the studied publications. The concept matrix will include an overview of current research streams and details of the considered publications. The overview of the research streams will reflect which contents and topics are covered in current research on DevOps. Thus, it will provide an overview of the current state of knowledge on this topic.

The second intermediate result represents a conceptualization of DevOps definitions. Therefore, the set of considered publications will be examined as well as the DevOps definitions that have been described will be collected. Based on this,

DevOps definitions will be provided. These will be developed and derived based on the examined publications. Additionally, the definitions will be conceptualized based on the derived concepts of the developed concept matrix. The goal of this step is to provide context-based DevOps definitions as an aggregation of the different research directions or domains affected by DevOps.

Third, a supportive decision framework for DevOps implementation is elaborated. The framework will be based on the first two intermediate results of this stage. Based on the research question, challenges and goals in the implementation of DevOps will be addressed. Thereby, influencing independent and dependent variables for this framework will be derived. The goal of the framework is to provide an overview of which line of DevOps definition should be utilized to implement DevOps.

4 Results

In the following chapter, the elaborated interconnected results are described based on the methodology defined in chapter *Methodology: Systematic Literature Review*. For this purpose, the results are described in three separate results parts. The first part of the results comprises the development of the current body of knowledge of DevOps. In this context, the elaborated concept matrix is presented, which categorizes DevOps topics. The second part comprises the derivation of DevOps definitions from the considered publications of the systematic literature review. In the third part, the developed supportive DevOps implementation decision framework and its components are presented.

4.1 Results I: Current Body of Knowledge on DevOps

Regarding current DevOps research, DevOps is a highly interdisciplinary research area that covers multiple topics as concepts, tools, and team structures (Hüttermann, 2021, pp. 1-4; Krey et al., 2022, p. 7307). This section presents the current body of knowledge on DevOps research. For this purpose, a concept matrix is elaborated. The results are based on the systematic literature review approach described within *chapter 3 Methodology: Systematic Literature Review*. As a result, an overview of the publications and relevant topics and details is given. Furthermore, the concept matrix derived several categories to conceptualize the studied DevOps topics within the considered publications which will be described in the following.

Table 10: Concept Matrix

				RESEARCH METHOD					TYPE OF PUBLICATION				DEVOPS PRACTICES							DEVOPS PRINCIPLES							DEVOPS GROUPS			DEVOPS MANAGEMENT					DEVOPS TOOLS														
ARTICLE (APA CTATION)	Year	Publication	Database	Statement	Case Study	(Systematic) Literature	Survey	Framework	Grounded Theory	Research Paper	Article	Proceedings	Short Paper	Continuous Delivery	Continuous Integration	Continuous Monitoring	Continuous Deployment	Continuous Learning	Microservices	Embedded Systems	Culture	Automation	Lean	Measurement	Monitoring	Sharing	Collaboration	Agile	DevOps Teams	Agile Teams	Cross-functional Teams	DevOps Engineers	Orchestration	Digital Transformation	DevOps for Large Scale	IT Governance	Maturity Models	GitOps	Build Tools	Automation Tools	Logging Tools	Monitoring Tools	Continuous Integration	Knowledge Sharing Tools	Deployment Tools				
Amaro et al., (2023)	2023	IEEE Trans on Software Eng	IEEE (Comp Soc & Xplore)			x				x				x	x	x	x				x			x																									
Bass, (2018)	2018	IEEE Software	ProQuest	x							x						x																																
Beetz & Harrer, (2022)	2022	IEEE Software	ProQuest			x					x			x	x				x						x												x												
Dornenburg, (2018)	2018	IEEE Software	ProQuest	x							x																		x																				
Ebert et al., (2016)	2016	IEEE Software	ProQuest	x							x			x					x	x	x																									x			
Gall & Pigni, (2018)	2018	AMCIS Proceedings	AISNET		x							x																																					
Galup et al., (2020)	2020	Communications of the ACM	EBSCOhost	x	x						x											x	x	x					x																				
Ghantous & Gill (2017)	2017	PACIS Proceedings	AISNET			x						x		x			x					x						x																		x			
Hemon et al., (2020)	2020	IEEE Software	ProQuest		x					x				x	x												x		x			x																	
Hemon et al., (2018)	2018	International Working Conference on Transfer and Diffusion of IT	Backward Search		x					x				x	x													x			x																		
Hemon-Hildgen & Rowe (2022)	2022	European Journal of IS	EBSCOhost	x							x			x			x																																
Hemon-Hildgen et al., (2020)	2020	European Journal of IS	EBSCOhost		x					x				x	x		x	x				x						x						x															
Hüttermann, (2021)	2021	ECIS Proceedings	AISNET		x							x		x								x	x				x																						
Hüttermann & Rosenkranz, (2019)	2019	ICIS Proceedings	AISNET			x							x																																				
Kersten, (2018)	2018	IEEE Software	ProQuest		x								x																																		x		x
Krey et al., (2022)	2022	HICSS Proceedings	Forward Search			x						x		x								x	x							x																x			
Leite et al., (2019)	2019	ACM Computing Surveys	ACM (Digital Lib)				x	x		x				x			x		x											x		x	x					x								x		x	
López-Fernández et al., (2022)	2021	IEEE Trans on Software Eng	IEEE (Comp Soc & Xplore)						x	x												x						x		x							x												
Lwakatare et al., (2016a)	2016	HICSS Proceedings	Forward & Backward Search		x							x		x			x			x																													

DevOps: A Supportive Decision Implementation Framework

Lwakatare et al., (2016b)	2016	ICSEA Proceedings	Backward Search	x						x								x	x		x	x																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
---------------------------	------	-------------------	-----------------	---	--	--	--	--	--	---	--	--	--	--	--	--	--	---	---	--	---	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Table 10 shows the elaborated concept matrix which states the current body of knowledge of the examined DevOps publications. In total, 33 publications were examined. The considered publications include 9 research papers, 10 articles, 12 conferences proceedings, and 2 short papers. All examined publications were published within a time range of seven years between 2016 and 2022, as seen in Appendix C. Thereby, the publications on DevOps were evenly published over the examined period, except a minor increase in 2018. Since DevOps is still a young research area (Jabbari et al., 2016, p. 1), this time range gives an overview of topics that are relevant to DevOps research in a timely manner. It can infer trends and emerging issues. Thus, a broad view of DevOps research and relevant topics is possible.

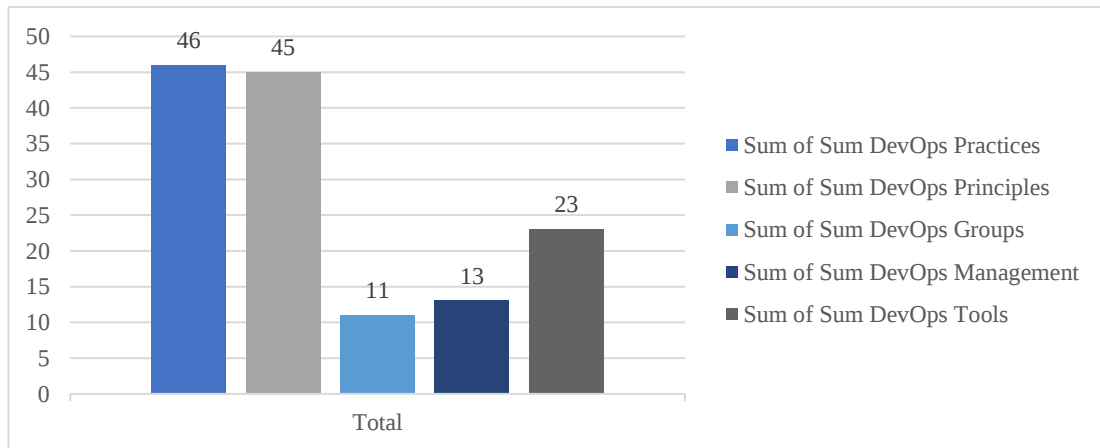


Figure 3: Total of referenced DevOps categories within the systematic literature review

Within the 33 publications, a total of 32 topics related to DevOps were considered. To categorize the 32 topics into categories, reference is made to DevOps categories which are based on logical summaries of topics and on categories that were described within the examined publications (for example, Beetz & Harrer, 2022; Ghantous & Gill, 2017; Hemon-Hildgen et al., 2020; Leite et al., 2020). This results in five DevOps main categories *DevOps Practices*, *DevOps Principles*, *DevOps Groups*, *DevOps Management*, and *DevOps Tools* within the concept matrix.

The category of DevOps Practices was the most referenced category with 46 publications. The DevOps Principles category was referenced 45 times, followed by DevOps Tools, which was referenced 23 times. The least referenced categories are

DevOps Management (referenced 13 times) and DevOps Groups (referenced 11 times).

The topics DevOps Practices and DevOps Principles were considered continuously and highly over the observed period. In comparison, the topics DevOps Groups and DevOps Management were not considered before 2018. In addition, these topic categories are less examined than DevOps Practices and DevOps Principles. In comparison to the timely trend of the categories DevOps Management and DevOps Groups, the publications initially had a focus on DevOps Tools, which decreased from 2020. In the following, the individual main categories of the concept matrix will be explained and considered regarding their trends and emerging issues. *Figure 3* shows an overview of the total numbers of referenced categories within all examined publications.

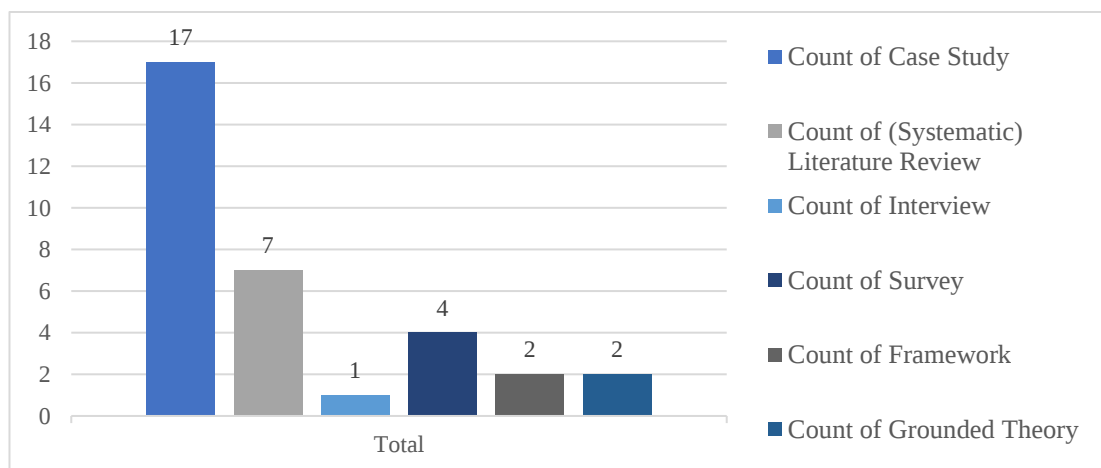


Figure 4: Research Methods of all considered publications

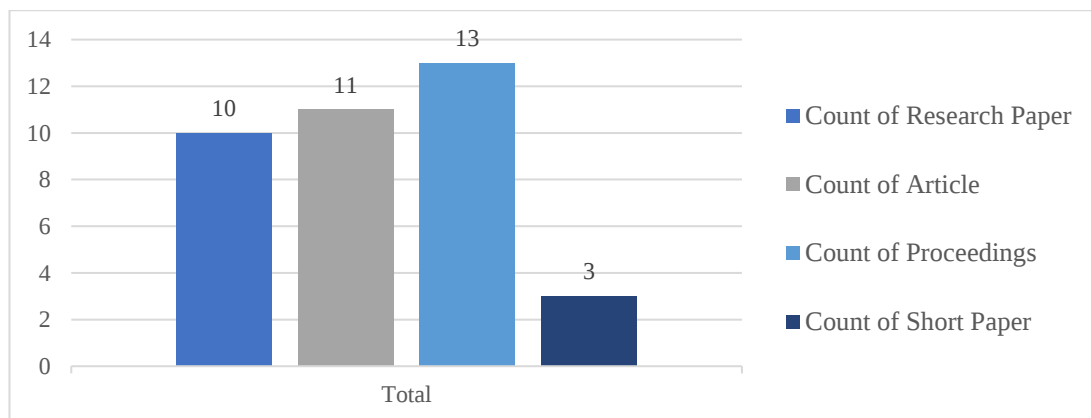


Figure 5: Publication types of all considered publications

As shown within Figure 4 and Figure 5, the categories show several trends regarding publication type and research method. Within the underlying publications, a trend towards practice-oriented publications can be observed. The publications follow 16 case study and 3 case study research approaches with a high practice orientation. Furthermore, this trend is particularly evident in the number of DevOps publications in practice-oriented journals such as the IEEE Software Journal. Furthermore, articles and publications can be considered as practice oriented. 8 of the 10 examined articles are published in the IEEE Software Journal. The underlying proceedings include 7 case studies and 2 surveys which leads to a practice-oriented focus within this publication type.

Within these two publication types, another trend can be identified. Most of the underlying articles mainly consider the DevOps Practices category. In addition, considered proceedings primarily address the categories DevOps Practices and DevOps Principles.

In comparison to proceedings and articles, considered research and short papers address a broader view on DevOps categories. Notably, these publications consider DevOps Practices and Principles, but also focus the other topic categories of DevOps Groups, Management, and Tools. This research can be seen as more diverse in terms of examined contents. This shows a possible imbalance within the more practice-oriented publications.

In the following, the elaborated categories are reviewed individually in more detail.

4.1.1: DevOps Practices

The category *DevOps Practices* refers to practices used by companies in the context of DevOps (Beetz & Harrer, 2022, p. 71; Ghantous & Gill, 2017, p. 4). These practices support and enable DevOps based iterative software delivery and workflows between development and operations (Leite et al., 2020, pp. 2-3). The category DevOps Practices summarizes the examined topics *Continuous Delivery*, *Continuous Integration*, *Continuous Monitoring*, *Continuous Deployment*, *Continuous Learning*, *Microservices*, and *Embedded Systems*. The DevOps Practices category was mentioned within 20 publications. In total, it was referred to 46 times within the considered publications with a dominating share of proceedings considering the category. Thereby, *Continuous Delivery* was the most referenced with 15 references,

followed by *Continuous Integration* and *Continuous Deployment* which were each referenced 10 times within the publications.

The topics on continuity, for example, Continuous Delivery, Continuous Integration, etc., are summarized by the term *continuity* for simplifying further descriptions and explanations. Continuity topics are particularly outstanding in the DevOps Practice category. Regarding all examined publications and categories, continuity topics are the most referenced topics. Additionally, these topics were studied evenly over the observed time period of 2016 to 2022. Also, as seen within *Table 10*, the publications considering DevOps Practices usually examine several DevOps Practices topics at a time, for example, continuous integration and continuous deployment or microservices and continuous delivery.

Taking a closer look at the stated implications within the publications, additional insights for the category DevOps Practices can be identified. In practical use, DevOps Practices are challenging for organizations. Thereby, pressure of continuity in development, integration, deployment, and delivery is identified. It is concluded that it in particular effects developers (Amaro et al.; 2022, pp. 1-3).

4.1.2: DevOps Principles

The category *DevOps Principles* describes underlying mentalities and mindsets for DevOps. Thereby, the principles set a scope for DevOps by providing guidelines for the implementation and maintaining of DevOps (Beetz & Harrer, 2022, p. 71; Krey et al., 2022, pp. 7297-7298). Thus, guided by conceptualizations of DevOps Principles of Beetz & Harrer (2022), Krey et al. (2022), and Stray et al. (2019), the principles include *Culture*, *Automation*, *Lean*, *Measurement*, *Monitoring*, *Sharing*, *Collaboration*, and *Agile*. These eight topics address different aspects of DevOps mindsets and values.

The category DevOps Principles was referred to within 19 publications. In total, it was referred to 45 times within the considered publications. Thereby, it was equally considered by research papers, articles, and proceedings. Leading topics within this category are represented by *Culture* and *Automation* which were referenced each 10 times within the publications.

The publications on DevOps Principles highly consider the cultural aspect of DevOps. In total, it is studied by 10 out of all 33 examined publications. Also, five of these publications are proceedings from conferences with a practical focus. Within

these publications, the authors refer to DevOps culture as a social movement to break down barriers between development and operations within an IT organization (Hüttermann, 2021, p. 2; Krey et al., 2022, p. 7306). However, the topic of DevOps culture belongs to the more frequently studied topics. In addition, this topic is also often considered in combination with several other DevOps Principle topics as well as DevOps Practice topics with a special focus on continuity aspects.

Furthermore, publications often study several topics on DevOps Principles at a time. In the case of the predefined CALMS term, this is particularly noticeable as these subjects are mainly acknowledged simultaneously by the publications (for example, Lwakatare et al., 2016b; Shropshire et al., 2017; Wiedemann et al., 2019). Additionally, collaboration is an important topic within this category. Various authors point out that collaboration is a fundamental foundation for the successful implementation of DevOps. It is a main influence for bringing together the two areas of development and operations (López-Fernández et al., 2022, p. 8; Rowse & Cohen, 2021, pp. 6785-6786). Moreover, this argument is mainly considered within proceedings and in combination with the topic of culture with a more practical focus.

In summary, DevOps Principles category is continuously referred to within research. Thereby, publications place particular emphasis on culture and collaboration.

4.1.3: DevOps Groups

The category *DevOps Groups* summarizes team structures that focus on fulfilling the amalgamation of development and operations (Wiedemann et al., 2020, p. 459). The DevOps Groups category includes *DevOps Teams*, *Agile Teams*, *Cross-functional Teams*, and *DevOps Engineers*. The category DevOps Groups was referred to within 8 publications. In total, it was referred to 11 times within the considered publications with a dominating share of research papers considering the category. Thereby, *DevOps Teams* was the most referenced with 5 references within the publications.

Considering the subjects within the publications, DevOps Groups represents a transformation of collaboration within organizations. Authors state the changing perception of IT into IT as a team activity (Amaro et al., 2023, pp. 2-3; Dornenburg, 2018, pp. 71-74). DevOps is perceived as an overarching IT team activity in which people from different areas work together (Dornenburg, 2018, p. 71). Thereby, this category primarily studies the underlying workflows and collaborations when

implementing DevOps (Lwakatare et al., 2016a, pp. 1-2; Wiedemann et al., 2019, p. 46). Notably, the topics within this category refer to the DevOps Principles topics of culture and collaboration. The implementation of DevOps requires team and IT organizations restructuring to achieve, for example, cross-functional workflows and improve the quality of the IS development in terms of releases and time. This poses several risks of these team activities that impact the success of DevOps within an organization (Dornenburg, 2018, pp. 71-74; Hemon-Hildgen et al., 2020, p. 9).

Based on this, different responsibilities and team structures can be identified in the DevOps Groups category. Thereby, agile teams also fall under the consideration of DevOps Groups category, as DevOps has historically grown from agile methodologies and practices. As described within the publications, agile teams partly contain cross-functionality and pursue similar goals as classic DevOps teams (Wiedemann et al., 2020, p. 459).

However, DevOps Groups are the least referenced category within the considered publications. Publications on DevOps Groups have only been noted since 2018 but are continuously published since then. Nevertheless, a consideration of DevOps Groups is important within research. DevOps Groups topics build the foundation for successful DevOps implementation (Hemon-Hildgen et al., 2020, p. 9).

The four topics within the category DevOps Groups are comparatively similar to each other by covering related aspects. These topics highly focus on collaboration within teams as well as independent and conscientious processing of tasks (Hemon et al., 2019, pp. 3-4; López-Fernández et al., 2022, p. 8).

4.1.4: DevOps Management

The *DevOps Management* category refers to tasks for implementing and maintaining DevOps within an organization. Implementing DevOps leads to new organizational challenges which involve new management and administrative tasks (Hemon-Hildgen et al., 2020, p. 12). The DevOps Management category covers the topics of *Orchestration*, *Digital Transformation*, *DevOps for Large Scale Approaches*, *IT Governance*, *Maturity Models*, and *GitOps*. Next to the category DevOps Groups, the category DevOps Management is the second least referenced category.

The category DevOps Management was referred to within 12 publications. In total, it was referred to 13 times within the considered publications with a dominating

share of research papers considering the category. Thereby, all topics were considered nearly equally.

The majority of DevOps Management publications are case studies with a practical focus. DevOps Management topics are considered either on its own or in combination with the category DevOps Practices. Regarding the perception of DevOps Management category to DevOps Practices, various publications refer to the use of practices to support management tasks within the DevOps context (Amaro et al., 2023, p. 4; Wiedemann et al., 2020, p. 466).

The topics in this category highlight the organizational challenges in DevOps context. In this context, the topic of *DevOps for Large Scale Approaches* refers to the increased management effort required by larger companies to implement DevOps (Hemon-Hildgen et al., 2020, p. 12). In addition, GitOps is a topic that is also considered in the publications. It refers to emerging trends in the DevOps movement (Beetz & Harrer, 2022, pp. 70-75).

Considering the publications subjects in depth, DevOps is stated as an ongoing trend that many organizations are trying to adopt (Amaro et al., 2023, p. 1; Hemon et al., 2018, p. 2). Thereby, several authors note that not all organizations are able to successfully implement DevOps (Amaro et al., 2023, p. 1; Gall & Pigni, 2018, p.1 ; Ghantous & Gill, 2017, p. 6; Rowse & Cohen, 2021, p. 6785). Publications on DevOps Management analyze these circumstances. Stated reasons for implementation problems include a lack of management information and skills, as well as a lack of shared understanding and vision, which make the process for some adopters more complicated (Amaro et al., 2023, pp. 1-2; Ebert et al., 2016, p. 100). In context of the DevOps Management category, authors point out the need for management guidance to achieve business goals and to improve IT systems performance (Amaro et al., 2023, p. 1).

4.1.5: DevOps Tools

The *DevOps Tools* category describes tool sets that support the implementation of DevOps Practices (Ebert et al., 2016, pp. 95-97). The use of tools in DevOps has enabled developers improved and faster collaboration (Dornenburg, 2018, pp. 71-73). The category DevOps Tools was referred to within 6 publications. In total, it was referred to 23 times within the selected publications. Thereby, *Monitoring Tools* was

the most referenced topic with a total of six references. In this context, publications considered several DevOps Tools topics simultaneously.

In general, the DevOps Tools category is considered primarily in conjunction with the DevOps Practices category. This supports the proximity of these topics. In the DevOps Tools category, various tool functionalities were considered such as *Build Tools*, *Automation Tools*, *Logging Tools*, *Continuous Integration Tools*, *Knowledge Sharing Tools*, and *Deployment Tools*.

In summary, these tools are often used to support the IS operations of an organization (Ebert et al., 2016, pp. 95-97). The publications in the category referenced DevOps Tools primarily from 2016 to 2018. However, the topic has been studied less since 2019. It is also noticeable that monitoring tools were considered within every publication that considered the DevOps Tools category.

4.2 Results II: DevOps Definitions

In this section, an overview of possible DevOps definitions as an aggregation of different lines of research, respectively domains, affected by DevOps is given. In the following, reference is made to the five defined categories from the concept matrix *DevOps Practices*, *DevOps Principles*, *DevOps Groups*, *DevOps Management*, and *DevOps Tools*. Based on identified DevOps definitions in the considered publications within the systematic literature review, five possible DevOps definitions are developed.

In summary, various DevOps definitions were provided within the 33 examined publications, as seen in *Appendix D*. 22 of the publications were based on DevOps definitions from previous research. These definitions were taken from primary sources or based on primary sources with adaptations and own reproductions.

Six publications provided their own DevOps definitions without references to primary sources. Those were developed by the authors based on their knowledge, assumptions, and common understanding of DevOps (for example, Ebert et al., 2016; Hemon et al., 2020; Hemon et al., 2019; López-Fernández, 2022). Notably, all six publications were research papers. Therefore, they are close to the DevOps research and resolution of the topic of DevOps while being less practice oriented. These six publications study the topic of DevOps within all defined categories from the concept matrix and do not, for example, only consider one category such as DevOps Practices. In addition, five of the considered publications did not provide a DevOps definition

and elaborated the topic DevOps in their publications based on a general basic understanding. These were mainly article publications that specifically highlighted a particular DevOps topic (for example, Dornenburg, 2018; Kersten, 2018; Ozkaya, 2020; Smart, 2018).

In the context of the 22 publications which used predefined DevOps definitions, *Table 11* shows a small overview of the most referenced DevOps definitions. The definitions of Bass et al. (2015) and Jabbari et al. (2016) were used by several publications. Furthermore, DevOps definitions were merged by several publications. Thereby, DevOps definitions of Dyck et al. (2015), Humble & Farley (2010) and Humble & Molesky (2011) were highly referred to.

Table 11: Common DevOps Definitions within DevOps research

Authors	DevOps Definitions
Bass et al. (2015)	DevOps is a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality.
Dyck et al. (2015)	DevOps is an organizational approach that stresses empathy and cross-functional collaboration within and between teams – especially development and IT operations – in software development organizations, in order to operate resilient systems and accelerate delivery of changes.
Jabbari et al. (2016)	[DevOps is] a development methodology aimed at bridging the gap between Development (Dev) and Operations, emphasizing communication and collaboration, continuous integration, quality assurance and delivery with automated deployment utilizing a set of development practices.

Note: This overview is restricted by the availability of the original sources of the considered DevOps definitions.

The DevOps definitions considered within research follow different approaches and focus different aspects which are subjectively relevant to the topic area of DevOps. Notably, a large proportion of the definitions have equated DevOps with principles and practices in the context of DevOps, as seen within *Appendix D*. Furthermore, the topics of cross-functionality, the bringing together of development and operations, and the improvement of software development were also increasingly considered within the definitions. In this respect, the definitions show parallels to the DevOps categories from *4.1 Results I: Current Body of Knowledge on DevOps*.

Therefore, the collected DevOps definitions were synthesized and assigned to the different DevOps categories within the context of this thesis.

Within *Table 12*, five DevOps definitions are provided in the considered areas of the concept matrix. These are based on the definitions of the examined publications within the systematic literature review and are logically compiled and elaborated.

Table 12: DevOps Definitions

Category	Definition	Source
DevOps Practices	DevOps is a set of practices “that streamline[s] the software implementation process” ⁽¹⁾ “and at the same time covers all the aspects which help in speedy, optimized and high-quality [continuous] software delivery” ⁽²⁾ .	(1) Hemon-Hildgen et al., 2020 (2) Ghantous & Gill, 2017 <i>Others:</i> Amaro et al., 2023 Bass, 2018 Ozkaya, 2019 Shorpsire et al., 2017 Zhu et al., 2016
DevOps Principles	“DevOps is a cultural movement” ⁽¹⁾ that is “aimed at bridging the gap between Development and Operations” ⁽²⁾ by setting a frame to their collaboration, communication, values, and goals.	(1) Krey et al., 2022 (2) Jabbari et al., 2016 <i>Others:</i> López-Fernández et al., 2022
DevOps Groups	DevOps refers to cross-functional and team working “that breaks complex architecture and features sets” ⁽¹⁾ “where each team member need[s] to consider and anticipate the job to be done by other members” ⁽²⁾ .	(1) Stray et al., 2019 (2) Hemon et al., 2019 <i>Others:</i> Ebert et al., 2016 Leite et al., 2020 Wiedemann et al., 2020
DevOps Management	DevOps “improves and accelerates the delivery of business value by making the collaboration between development and operations effective” ⁽¹⁾ and delivering guidance for DevOps implementation and maintenance.	(1) López-Fernández et al., 2022 <i>Others:</i> Ebert et al., 2016 Krey et al., 2022 Rowse & Cohen, 2021
DevOps Tools	DevOps describes the usage of several tools to support process automation within an IT organization to support Development and Operations.	Ebert et al., 2016 Lwakatare et al., 2016a Wiedemann, 2018

Note: Own representation

Based on the definitions of the individual categories, a general DevOps definition can be derived. This should contain all the important components for the DevOps topic area. A DevOps definition, which covers all important components could look as follows:

DevOps is a cultural movement requiring leading management with the aim to break down barriers between development and operations and to emphasize collaboration and communication to optimize IS development processes by automation and tool usage (Ebert et al., 2016, p. 94; Jabbari et al., 2016, p. 6; Krey et al., 2022, p. 7299; López-Fernández et al., 2022, p. 1).

In the following, the individual DevOps definitions are considered based on their respective category.

4.2.1: DevOps Practices

This DevOps definition centers on practices to optimize the process of software development and deployment. Within the considered DevOps definitions, DevOps was more overly equated to DevOps Practices. Thereby, a DevOps Practices definition is derived. The stated definition summarizes underlying DevOps Practices definitions and contains their necessary components (Beetz & Harrer, 2022, p. 71).

This stated definitions' focus is on improving IS development process efficiently. Thereby, the DevOps Practices definition relies on the aspect of continuity in DevOps, as there is a transition to continuous releases (Hemon-Hildgen & Rowe, 2022, p. 568). Processes should be optimized in terms of quality and speed. This definition illustrates the possibilities of DevOps within an organization.

4.2.2: DevOps Principles

The definition of DevOps Principles emphasizes DevOps as a cultural shift and a new way of thinking about Development and Operations. Based on underlying definitions that focus DevOps Principles, this definition is derived. Considered DevOps definitions highly considered DevOps Practices and Principles (Beetz & Harrer, 2022, p. 71). Also, a focus on equating DevOps to the CALMS principles could be observed (Wiedemann et al., 2019, pp. 46-48).

The goal of this stated definition is to spread the values behind DevOps and to highlight the relevance of DevOps within an enterprise context. It sets a scope for the DevOps mindset in terms of communication, collaboration, value, and goals.

4.2.3: DevOps Groups

The definition in the DevOps Groups section refers to the aspect of cross-functional collaboration in the DevOps context as well as the importance of the individual team members in an organization that implements DevOps.

Within all considered DevOps definitions, a focus on the relevance of collaboration could be observed (Beetz & Harrer, 2022, p. 71; Hemon-Hildgen et al., 2020, pp. 9-10). The goal of this stated definition is to strengthen the individual groups and their collaboration with each other in order to show their relevance. Therefore, the focus is also on the personal responsibility and accountability of the team members.

4.2.4: DevOps Management

The definition of DevOps in the DevOps Management concept focuses on the value that a DevOps implementation can have for an organization. The goal of this definition is to build an understanding of the role of management in the DevOps context and its ability to lead it.

However, the considered definitions rarely focus the management aspect of DevOps. Thereby, they focused more overly the value delivering aspects of DevOps.

4.2.5: DevOps Tools

The definition of DevOps in context of DevOps Tools category centers the use of tools to support the areas of development and operations. The goal of this definition is to highlight the value of tools in DevOps implementation. It also refers to process automation that supports the DevOps approach (Lwakatare et al., 2016b, pp. 94-95).

Furthermore, it has to be noted that DevOps definitions rarely consider the DevOps tools category. Thereby, the considered DevOps definitions increasingly focus on process automation, which is includes by DevOps tools (for example, Hemon-Hildgen et al., 2022; Marnewick & Langerman, 2021; Sharp & Babb, 2018).

4.3 Results III: Supportive DevOps Implementation Decision Framework

In this section, a supporting DevOps implementation decision framework is derived based on the underlying research question. As shown in Figure 6, the framework provides an overview of which line of DevOps definition should be utilized

to implement DevOps. Thereby, the definitions are related to respective goals and given challenges in IS development and target groups which were stated in the theoretical background.

The framework is separated into three main parts. First, challenges and goals in IS development are presented. These goals and challenges are separated into five distinct areas *Quality Information Systems*, *Competitive Advantages*, *Information Systems Environment*, *DevOps*, and *Effective Collaboration*. Thereby, DevOps challenges and goals are included into the framework regarding the research question and its focus on DevOps implementation. The second part is represented by DevOps definitions of the results part two. They will be matched the described challenges and goals. Furthermore, the definitions are connected to specific target groups. The target groups address where to locate the chosen DevOps definitions to tackle given challenges or achieve respective goals.

4.3.1 Challenges and Goals of IS Development and DevOps Implementation

The framework identifies five areas of challenges and goals in modern IS development derived by stated challenges and goals of *2.1 Information Systems Development*. Thereby, these challenges and goals represent the independent variables of the elaborated framework.

4.3.1.1 Quality Information Systems

The *Quality Information Systems* area addresses the challenges and goals that IS development faces with respect to the underlying quality of its software products. The aim of this area is to describe the need of companies to implement systems that are sustainable for the organization. This includes the longevity and automation of systems such that economic effort can be reduced (Amaro et al., 2023, p. 1; Sommerville, 2020b, pp. 16-26).

To support these challenges and goals with a DevOps implementation, the categories DevOps Tools and DevOps Practices are of particular relevance. These focus on the optimization and automation of software development processes and can thus promote an increase in the quality of IS.

4.3.1.2 Competitive Advantages

The area of *Competitive Advantage* describes the condition of the current IS market. In general, the competitive pressure has increased due to the fast pace and the

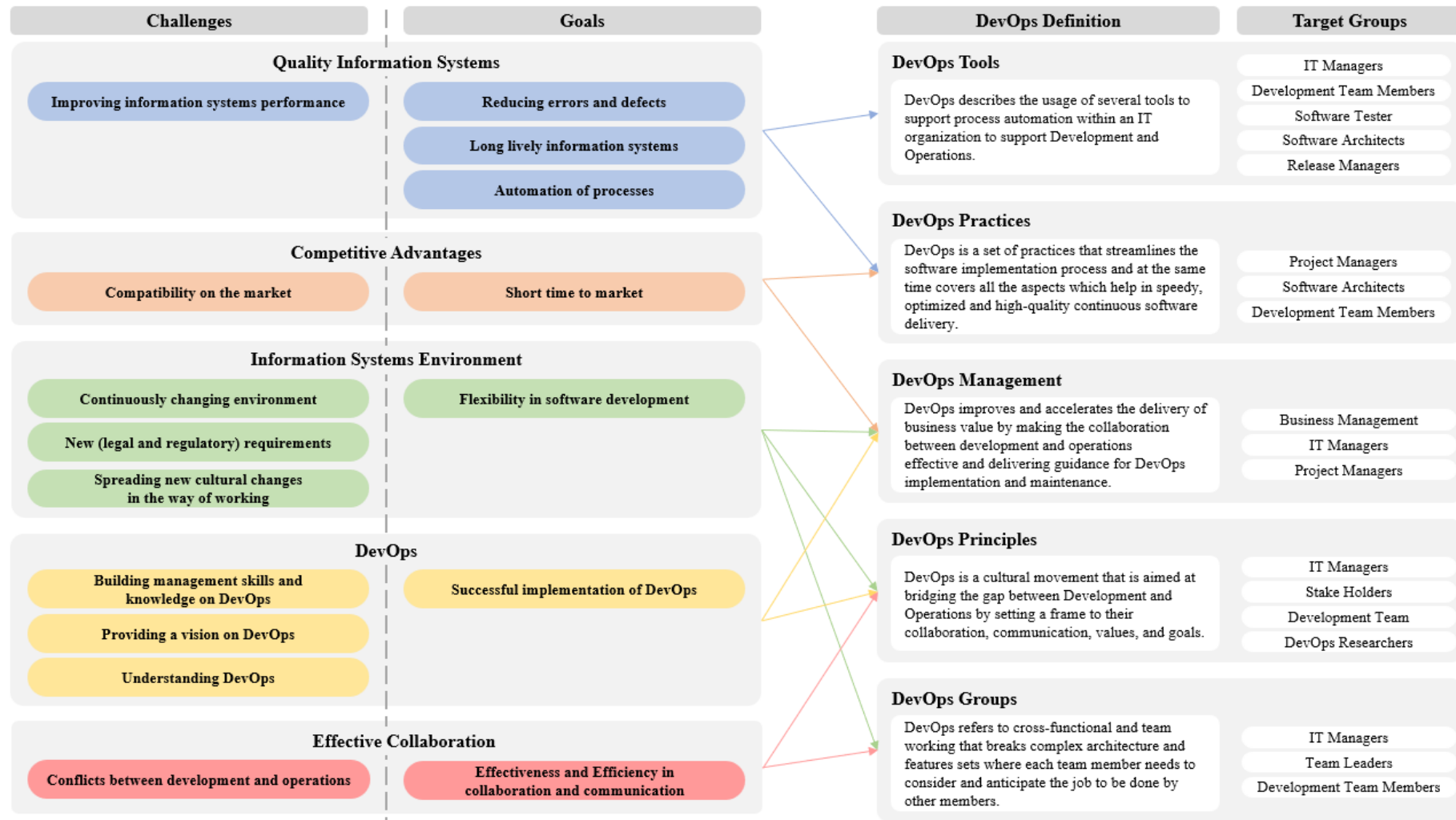


Figure 6: Supportive DevOps Implementation Decision Framework

pressure to innovate. Due to increasing digitalization, organizations are now subject to additional challenges. They are pressured to release faster and be more flexible in their IS development processes in order to maintain their position on the market in the long term (Amaro et al., 2023, p. 1; Sommerville, 2020b, pp. 16-26; Sommerville, 2020a, pp. 11-14).

A DevOps implementation can support these challenges and goals. Thereby, the categories DevOps Practices and DevOps Management can be considered. On the one hand, DevOps Practices show the practices and approaches for optimizing processes so that competitive advantages can be created. On the other hand, this can be supported by DevOps Management. The DevOps Management category highlights the value of a DevOps implementation. In this way, a fundamental basis for a DevOps implementation can be created in order to achieve competitive advantages.

4.3.1.3 Information Systems Environment

The area of *Information Systems Environment* describes current challenges and goals that influence the environment of software development. Thus, the challenges focus the circumstance of changing requirements in software development and changing team structures. Consequently, the whole process of software development is influenced by these challenges (Sommerville, 2020b, pp. 16-26; Sommerville, 2020a, pp. 11-14). The underlying goal is on flexibility in software development. This is required by the changing and new requirements which dominate the perception of information systems environment.

In general, DevOps implementation can help address these challenges and achieve these goals. This requires a top-down consideration of DevOps within an organization. Thereby, DevOps Management, DevOps Principles, and DevOps Groups are of relevance. These three categories support responsiveness and flexibility at different complexity levels within organizations.

4.3.1.4 DevOps

The area of *DevOps* is additionally considered in the challenges and goals. Based on the underlying research question, DevOps implementation also faces challenges and goals within the process of IS development. In general, DevOps implementation requires understanding for the topic area of DevOps. Therefore, the challenges and goals of this area address the fundamental understanding and skills of DevOps within an organization, in order for DevOps to be implemented successfully

(Amaro et al., 2023, pp. 1-2; Dornenburg, 2018, pp. 73-74; Hemon-Hildgen & Rowse, 2022, pp. 569-570; Hüttermann, 2021, p. 4). The categories DevOps Management and DevOps Principles are of relevance in the context of goals and challenges in DevOps used within practices. They promote DevOps understanding and skills.

4.3.1.5 *Effective Collaboration*

Effective Collaboration describes the need for collaboration and good communication. The successful implementation of systems depends on the collaboration of teams, team members and other necessary cross-functional relationships (Amaro et al., 2023, pp. 1-2; Dornenburg, 2018, pp. 73-74). This can be promoted by the categories DevOps Principles and DevOps Groups as they convey and encourage the value of collaboration and communication.

4.3.2 **DevOps Definitions and Target Groups**

Within the framework, target groups were assigned to the various DevOps categories. In general, the target groups are based on the roles identified in the theoretical background 2.1.4 *Modern Information Systems Development*. They should help to define addressees for the individual categories in order to simplify the applicability of the framework. In the following, the relationships and context between target group and DevOps definitions is considered individually for the different categories.

4.3.2.1: *DevOps Practices*

The definition of DevOps Practices focuses the optimization of software development and delivery processes. Thus, it addresses *product managers*, *software architects* and *development team members* in particular.

Implementing DevOps regarding DevOps Practices especially influences underlying processes (Beetz & Harrer, 2022, p. 71). Thereby, product managers and software architects oversee managing and implementing those changes. DevOps Practices definition supports these two target groups by aiming the advantages of DevOps implementation. Moreover, the definition is relevant to development team members. These are especially affected by changes of processes. DevOps Practices definition supports their understanding for these changes.

4.3.2.2: DevOps Principles

The definition of DevOps Principles refers to the value and relevance of implementing DevOps. This definition addresses *IT managers, stakeholders, development teams* and *DevOps researchers*.

The definition of DevOps Principles is focusing on fostering DevOps communication and reducing barriers between development and operations. It gives a frame to the underlying DevOps mindset and values (Rowse & Cohen, 2021; pp. 6788-6789; Stray et al., 2019, p. 7009). Regarding the considered target groups, DevOps Principles definition supports the understanding and implementation for DevOps. Therefore, given challenges and respective goals in the area of *Information Systems Environment, DevOps*, and *Effective Collaboration* can be addressed by this definition and by applying them to the target groups.

4.3.2.3: DevOps Groups

The definition of DevOps Groups refers to the responsibilities and accountability that individual groups assume in the DevOps context. Furthermore, the collaboration is addressed in regard to cross-functional and team activities. The definition addresses *IT managers, team leaders*, and *development team members*. The definition is especially important for the stated target groups as it emphasizes collaboration and responsibilities within the context of DevOps. Thus, it can point out the relevance to this target group.

4.3.2.4: DevOps Management

The definition of DevOps Management refers to the value of implementing DevOps and how management positions itself in relation to it. This definition addresses *Business Management, IT Management* and *Product Management*.

Regarding the target group, this definition is referring to management tasks and responsibilities within all stages of management in an organization. As management is responsible for successfully implementing DevOps, it is necessary to these target groups to understand the value of DevOps implementation to tackle given challenges and achieve respective goals (Amaro et al., 2023, pp. 1-2).

4.3.2.5: DevOps Tools

The definition of DevOps Tools refers to the use of tools to automate processes in software development and deployment. This definition centers target groups

regarding their development and operations tasks (Hemon-Hildgen et al., 2020, pp. 7-10; Leite et al., 2020, p. 13). These target groups include *IT managers, development team members, software tester, software architects, and release managers*.

5 Discussion

Being only 15 years old, DevOps is still a new trend in research and practice. Thus, DevOps is subject to continuous change and further development. In recent years, various trends have emerged based on the topic area of DevOps. Thereby, a lack of a clear definition of DevOps characterizes this area and allows development in the idea of DevOps (Beetz & Harrer, 2022, pp. 70-71).

Regarding current DevOps research, DevOps is a highly interdisciplinary research area that covers multiple topics such as concepts, tools, and team structures (Hüttermann, 2021, pp. 1-4; Krey et al., 2022, p. 7307). Within this thesis, the diverse topic area of DevOps is analyzed based on state-of-the-art literature. An overview of this research area is elaborated and categorized. Thereby, the results of this thesis contribute to the DevOps research area and point out important research streams.

Furthermore, the findings exhibit a practice-oriented focus of the research area. This results of the amount of examined articles and proceedings and followed research approaches such as case studies and interviews with a practical orientation. DevOps is a topic that is implemented within practice, which ties the observed research and its implications for this thesis into a practical context. Consequently, the results of this thesis can help the research community reshape the focus of their research.

In the following, the three parts of the results and the overarching conclusions are discussed based on a contribution to the research field and practice as well as their implications and limitations. Furthermore, a short overview of further general implications and limitations is given.

5.1 Results I: Current Body of Knowledge on DevOps

The current state of knowledge in the DevOps literature covers various aspects and research directions. This knowledge is conceptualized within the elaborated concept matrix, as seen in *Table 10: Concept Matrix*. Thereby, five individual DevOps categories are derived from observed DevOps research. Various trends can be identified regarding each identified DevOps category, which are relevant for future research. In the following, the identified categories in context of the developed concept

matrix will be discussed and implications and limitations for further research and practice will be derived. Furthermore, a consideration of merging categories within future DevOps research will be taken. In addition, a more practice-oriented viewpoint of DevOps research will be discussed.

5.1.1: DevOps Practices

The topic of DevOps Practices has shaped DevOps research in the past and continues to do so today (Beetz & Harrer, 2022, pp. 70-75). DevOps Practices category was 46 times referred to within the examined publications. . It also is evenly considered within the observed time period. Thus, this category is the most referenced within the results of the systematic literature review and can be assumed to keep relevance for DevOps research.

Within the considered publications, it has a special focus on continuity. Moreover, the publications point out a problem of understanding and applying DevOps Practices in practice which is assumed to be the reason for the lacking usage of DevOps Practices within organizations (Lwakatare et al., 2016b, p. 92). This practice-based problem can require a deeper focus on DevOps Practices within research resulting in an enhanced understanding of this areas application and understanding. Further research on this topic can be assumed to support the comprehensibility and classification of DevOps Practices, which helps to improve an organizations implementation of DevOps Practices.

The topic of continuity is prominent within the category DevOps Practices. It is highly referred to and considered within nearly all publications on DevOps Practices. Based on this, it can be assumed as a highly important topic area within DevOps Practices and DevOps research in general. Continuity represents the topic area of continuous integration, deployment, delivery, learning, and monitoring. It represents the transformation of developing and distributing information systems through implementing DevOps. This change consists of a shift from big one-time releases to iterative, feedback-based, and continuous releases. Nowadays, this can already be seen in practice (Beetz & Harrer, 2022, p. 75; Dornenburg, 2018, pp. 71-73). This transformation in software releases could be a reason for the increased attention on DevOps Practices. Furthermore, it implies a focus on the associated continuity topics within practice and research.

Simultaneously to the general DevOps definition problem, a definition problem could be identified within DevOps Practices. In the conceptualization of DevOps Practices, considered publications summarize DevOps Practices differently. Thereby, they have different focuses within their approaches, as seen in *Table 10: Concept Matrix*. Thus, no common understanding of DevOps Practices could be identified within DevOps research and practice (Hemon-Hildgen et al., 2020, p. 9; Lwakatare et al., 2016b, p. 92). Consequently, there is also no uniform definition and classification of the DevOps category DevOps Practices. It can be assumed that further research on DevOps Practices can help to conceptualize this topic category. In this thesis, the topic of DevOps Practices was summarized based on the identified topics and existing conceptualizations. Nevertheless, a more in-depth conceptualization of DevOps Practices in future research may further fan the topic, as this thesis' underlying summary of the category is dependent on the considered publications. This dependence could be assumed to restrict the results as the publications were selected with a wider consideration of the topic area of DevOps. Consequently, a consideration of DevOps Practices conceptualization with a focus on definitions of DevOps Practices within practical use can be proposed. This can be assumed to counteract the practical application problem of DevOps Practices within practice.

5.1.2: DevOps Principles

Similarities to DevOps Practices category can be observed in the category of DevOps Principles. DevOps Principles was referred to 45 times within the examined publications. This category is represented evenly within the observed time period. Therefore, DevOps Principles category can be assumed to remain in relevance within further DevOps research. DevOps Principles category sets the frame for DevOps by characterizing the area regarding its mindset and values. Furthermore, DevOps Principles give guidance for DevOps implementation. In addition, an understanding problem can also be observed within this category. No uniform definition and conceptualization of the category DevOps Principles could be identified within the considered publications, as seen in *D. Appendix: DevOps Definitions Overview*. It can be assumed that further research on DevOps Principles can help to conceptualize this topic category as this can help structuring this specific research stream.

Considering the research term CALMS within the context of DevOps Principles category, it is increasingly referred to within the examined publications (for

example, Amaro et al., 2023; Lwakatare et al., 2016b; Wiedemann et al., 2019). On the one hand, CALMS is referred to as a fundamental value set for DevOps and is often equated to DevOps in general and DevOps Principles category. Thereby, it could be assumed to miss out on important topics that are relevant to DevOps Principles. On the other hand, publications conceptualize DevOps Principles and only include part of CALMS components such as culture and automation into this category (for example, Krey et al., 2022; Ozkaya, 2019).

In the context of this thesis, the conceptualization of DevOps Principles follows the approach of CALMS extended by additional principles of DevOps. With this approach, the best possible compilation of the DevOps Principles category is intended. This provides a broad overview and does not limit the research results regarding a particular CALMS perspective. These diverse aspects in the DevOps Principles category do, however, overlap with other categories. The collaboration and cultural change aspect, for example, has parallels with DevOps Groups and DevOps Management.

In addition to collaboration, consideration of culture within DevOps Principles is of particular relevance. DevOps culture can be assumed to have a special importance within DevOps research and practice. Presumably, this could be based on the cultural transformation within organizations through the effect of an implementation of DevOps. This includes new ways of working and thinking as well as delivering and providing better products and service (Lwakatare et al., 2016b). The increasing consideration of DevOps cultures may indicate a need to better structure and promote understanding of DevOps at a holistic level including, for example, employees and management. In this context, several other areas can be considered. For example, employees and management are also affected by the structural changes of DevOps implementation. To promote the understanding of cultural changes, a future consideration of culture in DevOps research should be taken. Thereby, DevOps Management and DevOps Groups aspects should also be taken into consideration from a practical viewpoint. This can enable challenges in DevOps application to be highlighted in more detail.

In general, DevOps Principles category needs to be conceptualized in the same way as DevOps Practices. Both categories are already in use terms and are highly referenced namely within DevOps research and practices (for example, Beetz &

Harrer, 2022, p. 71; Ghantous & Gill, 2022, pp. 4-5). This could foster a common understanding of these two categories and of DevOps in general.

5.1.3: DevOps Groups

DevOps Groups can be considered as an emerging subject within DevOps research. With only eleven references within the examined publications, DevOps Groups represents the least referenced DevOps category. However, DevOps Groups category gained popularity within the end of the considered time period. While the initial focus of DevOps research was on DevOps Tools and improving organizations through its usage, the perception of DevOps changed. Nowadays, DevOps is characterized by organizational transformations in terms of their fundamental collaborations. This implies a change in the perception of DevOps towards a joint task area.

DevOps implementation largely influences development teams and how they collaborate within the work environment (Hemon et al., 2020, p. 2; Wiedemann et al., 2019, p. 49). Employees and development team members are parties most affected by the transformation to DevOps-driven IT (Shorpsire et al., 2017, pp. 1-2). Publications like Wiedemann et al. (2019) and Hemon-Hildgen & Rowe (2020) already consider the need of studying this research area. These transformations are challenging for organizations. Therefore, it can be assumed that this will lead to a future focus on DevOps Groups and guide future research.

The current body of knowledge on DevOps Groups focuses collaboration and cross-functional team activities as the main leading influences, as seen in *Table 10: Concept Matrix*. These topics are mainly considered jointly with DevOps Groups category. Thus, DevOps Groups can be assumed to be challenging regarding to these two aspects. Considering these circumstances, DevOps Groups could further be assumed to influence DevOps research. Regarding the changes an organization has to manage when implementing DevOps, this topic is of particular importance for a successful DevOps implementation. Therefore, a practical focus within the DevOps research is recommended to follow.

5.1.4: DevOps Management

DevOps Management category is the second least referenced category. Thereby, it is referenced 13 times within all considered publications. This category is also emerging and became more prominent within the last part of the considered time

period. A transformation from DevOps Tools with a more practical viewpoint to a more cultural and transforming perception of DevOps is also noticeable here. For DevOps research, this transformation, which is also seen in DevOps Groups category, can indicate a future change to a more fundamental consideration of DevOps from a top-down perspective. This can implicate more practice-oriented DevOps research in the future.

The category DevOps Management is the most wide-ranging category in terms of its application area. In comparison to the other categories, it was not conceptualized by examined literature as a category. Thereby, publications referred to management and administrative tasks in the context of DevOps Management category. Thus, the category is conceptualized within this thesis. However, DevOps Management topics are rarely considered in depth. Considering DevOps Management from a practical viewpoint, notably, management is highly required by DevOps implementation and influences a whole DevOps implementations' success (Ozkaya, 2019, pp. 4-5; Wiedemann et al., 2019, pp. 48-49). In terms of the impact of DevOps Management on an organization, this topic area is assumed to be of high importance from a practical viewpoint. Thus, the need for a stronger consideration of a DevOps implementation in terms of challenges and risks from a management perspective can be assumed.

5.1.5: DevOps Tools

In comparison to the other discussed categories, DevOps Tools category became less prominent within the DevOps research area. It is referred to 23 times within the examined publications. Initially, DevOps community considered DevOps Tools equal to DevOps. This perception changed over the considered time period and DevOps Tools lost attention from DevOps researchers.

This change could be assumed to have various reasons. As DevOps is still young within research and practice, benefits and impacts of DevOps implementation for an organization were initially measurable by the usage of DevOps Tools. From the management perspective, measurability is important when introducing new concepts and their effectiveness on an organization (Wiedemann et al., 2019, p. 47). DevOps Tools fulfill the management criteria to be measurable regarding their success. Additionally, DevOps Tools were introduced to also support the IS operations side. Therefore, this initial focus on DevOps Tools within the research could be explained.

However, a shift towards more fundamental DevOps challenges was taken within the publications and DevOps Tools attention decrease can be assumed to be based on these circumstances. In addition, it can also be assumed that DevOps Tools knowledge is more widely spread and therefore, its attention decreased.

Furthermore, a proximity towards DevOps Practices category within DevOps Tools category is observed. Based on similar scopes, this may lead tool areas being considered in the context of DevOps Practices in the future. In summary, DevOps Tools attention decreased within research and is not expected to gain in high relevance again.

5.1.6: Merging DevOps Categories

Considering the current body of knowledge of DevOps literature, DevOps Practices and DevOps Principles have a predominant focus. The examination of the categories DevOps Groups, Management and Tools is comparatively smaller. Nevertheless, one can see a proximity between these individual areas.

The areas DevOps Practices and DevOps Tools have a proximity based on their task areas of supporting IS development processes. With a bottom-up perspective, these two categories summarize used practices and tools. Thus, their goal is to support the daily work in the context of DevOps. However, the topic of DevOps Tools is observed with a decreased interest within DevOps literature. Nevertheless, tools are important to optimize and simplify processes in DevOps Practices (Ebert et al., 2016, pp. 94-98; Galup et al., 2020, pp. 48-53). Therefore, a proximity between these two categories can be assumed. DevOps Practices deal with the problem of proper application within organizations (Lwakatare et al., 2016b, p. 92). This could be counteracted by a joint consideration of DevOps Tools and DevOps Practices. Accordingly, considering them together may result in other research approaches, such as a focus on successful implementation of DevOps Practices by using DevOps Tools. Although there is no uniform approach to DevOps implementation (Hemon-Hildgen et al., 2020, p. 9; Lwakatare et al., 2016b, p. 92), organizations can be supported by studies in this area and thus minimize risks in the implementation of DevOps.

Furthermore, a closer look at the areas Principles, Management and Groups is taken. In comparison to the bottom-up perspective, these three categories deal with the DevOps topic from a more profound perspective. Regarding a top-down perspective within these categories, they focus the values and mechanisms to embrace DevOps

culture and thinking. Considering these categories jointly, this can simplify the implementation of DevOps within organizations. The categories can be summarized into a DevOps mindset which considers collaboration and communication. Thereby, collaboration and communication are of particular interest. Based on the results of the systematic literature review, as DevOps can only be implemented or improved in organizations by strengthening these two factors (López-Fernández et al., 2022, p. 8; Rowse & Cohen, 2021, pp. 6785-6786). In this context, various perspectives can also be considered, such as already done by Hemon-Hildgen & Rowse (2020) or Wiedemann et al. (2019). The authors focus DevOps in the context of management and employees and how to intensify this mindset. An overarching view of the three categories could provide a more holistic view on DevOps within organizations. This could counteract the problem of management lacking in information and skills in this field. At the same time, improved management could have positive impact on employees, employee satisfaction, and the efficiency and effectiveness of collaboration. This joint consideration allows considering interrelationships and effects between these application areas. Therefore, risks can be minimized. Furthermore, this allows a sustainable implementation of DevOps by following holistic approaches.

5.1.7: Practice-oriented DevOps research

DevOps has a strong connection to practice. Therefore, it cannot be separated from a practical viewpoint (Beetz & Harrer, 2022, p. 70; Hemon-Hildgen & Rowe, 2022, pp. 568-569; Hüttermann, 2021, p. 2). Based on the results of the systematic literature review, DevOps research is instrumental in ensuring that DevOps can be better implemented within organizations (ibid.). The two aspects of communication and collaboration are covered within all categories identified within the concept matrix. Based on this and on the underlying idea of DevOps, the aim is to bring development and operations closer together and break down its barriers.

Regarding DevOps being practiced in organizations, DevOps must be analyzed from the research perspective with a practical focus. This can lead to improvements in DevOps implementation and minimizing risks. Thus, DevOps should be detached from the pure research perspective, as is already the case in many of the considered publications (for example, Lwakatare, 2016b; Wiedemann, 2018). Considering various organizations and application areas in the context of DevOps can allow future

research to take a more unified view of the topic and identify patterns. This potentially can counteract the uniform definition problem of DevOps. This is especially important as the topic of DevOps is already strongly followed within practice.

Regarding the definition problem, a more profound relevance of practice in DevOps research can also enable the definition of DevOps to be consolidated and established from this perspective. In the long term, this approach could enable the topic to be further developed. Thereby, risks of DevOps implementation could be minimized.

5.2 Results II: DevOps Definitions

DevOps is characterized by a definition and common understanding problem. In this context, this thesis also identifies the lacking common definition and understanding of DevOps within DevOps research. Thereby, the definitions of DevOps within the considered publications address different topics within their individual definitions. This shows how varied the topic of DevOps is in research. Consequently, this also leads to a definitional problem in the practical application of DevOps. Within the publications' results, it quickly becomes apparent that the lack of a DevOps definition affects both practice and research (Amaro et al., 2023, p. 1; Beetz & Harrer, 2022, p. 70; Gall & Pigni, 2018, pp. 1-2; Ghantous & Gill, 2017, p. 1; Hemon-Hildgen & Rowse, 2022, p. 9; Hüttermann, 2021, pp. 1-2). On the one hand, the lacking definition allows DevOps elaboration. For example, new DevOps-based trends have emerged, and researchers are less restricted to a specific research direction (Amaro et al., 2023, pp. 14-16; Hemon-Hildgen & Rowse., 2020, p. 51; Hemon-Hildgen et al., 2022, pp. 568-573). On the other hand, the lack of a DevOps definition results in a missing common understanding in practice and research (Hemon-Hildgen et al., 2022, pp. 568-573; Hüttermann, 2021, pp. 6-7). In practice, this impacts organizations in several ways. As a result, organizations often lose the scope for their DevOps adoption, leading to a lack of management, vision, and success (Amaro et al., 2023, p. 1; Beetz & Harrer, 2022, p. 70; Gall & Pigni, 2018, pp. 1-2). Hemon-Hildgen & Rowe (2022) relate this circumstance to the complexity and newness of DevOps. They point out the need of tailoring the definition of DevOps to a specific context (Hemon-Hildgen & Rowe, 2022, p. 569; Hüttermann, 2021, pp. 5-11).

The definition problem is characterized by the diversity of the research domain (Hüttermann, 2021, pp. 1-4; Krey et al., 2022, p. 7307). Based on the results of the

systematic literature review, it is observed that the current state of research on DevOps is wide-ranging. It is not limited to a single topic category and requires a variety of DevOps definitions. Thus, the definitions within the examined publications pursue different approaches. The individual definitions are based on already provided definitions by various publications. Thereby, the individual definitions are adapted to a specific context of the DevOps domain.

Within this thesis, the approach of context specific DevOps definitions is followed and conceptualized into the five defined categories. Moreover, the definitions are based on the considered publications of the systematic literature review. Since the elaborated and derived DevOps definitions are based on the systematic literature review, they support the described diversity of the DevOps domain. In addition, the individual definitions are proximate to DevOps domains application, needs, and goals of DevOps in practice. This is based on the high practical focus of the considered publications which guides the results of this work. These application areas may vary and be based on different causes. Therefore, it is important to study different definitions, as it is done within this thesis.

In the context of this work, the examined publications of the systematic literature review are used to develop DevOps definitions that reflect the research field of DevOps. In doing so, the DevOps definitions reflect the five main categories of *Practices, Principles, Groups, Management, and Tools*.

This approach of classifying DevOps definitions into the different categories addresses current research streams of DevOps. There is a number of studies which tackle the definition problem of DevOps (for example, Hüttermann, 2021). Within the considered publications, definitions are characterized by their scope of application. Thus, the publications only consider isolated topics of DevOps. Consequently, the definitions were directly assigned to a specific area and purpose. However, this thesis follows a research-based approach that conceptualizes important research streams. This enables a holistic consideration of this research area and its influencing factors. Previous works that refer to DevOps definitions focus on individual aspects of the topic area DevOps.

This work builds on these underlying conditions of the DevOps definition problem. The goal is to structure the definitions based on their scope. Thus, for each application area, a definition is provided that pursues its categories' goals. In this way,

an overview of application specific definitions is provided. Thus, the definition problem is counteracted by limiting the generation of just single DevOps definitions.

The development of an overview of context-specific definitions is supported by previous research (for example, Hüttermann, 2021, pp. 6-7; Lwakatare et al., 2016b, p. 92). There, it is pointed out that DevOps definitions must be considered context specific. Within this thesis, various DevOps definitions were elaborated and derived from research-based DevOps definitions (as seen in *Table 12: DevOps Definitions*). By categorizing the definitions within the identified research streams, this thesis' DevOps definitions are based on this context-specific approach. Based on this, DevOps community can choose their individual definitions based on the needs and requirements of their context-specific DevOps application. The classification of definitions within this thesis allows to counteract the definition problem in the DevOps topic area by explicitly providing definitions for various research areas of the DevOps domain. By applying definitions in a specific context and structuring them, the needs of research and practice can be met.

As part of this thesis, a general DevOps definition was also developed that includes all relevant identified research streams and factors of DevOps. Several attempts have already been made in research to establish a uniform definition. These are limited to certain contents, as already described. However, the generally valid DevOps definition that has been developed should only be considered in connection with the individual definitions, as this ensures contextual dependency.

In the event of further DevOps definition research, a practical focus is recommended since DevOps has emerged from practice and is practiced there on a daily basis. This aims to develop a DevOps definition or set of DevOps definitions based on a common understanding within practice. Thus, this results in improved DevOps research due to its proximity to practiced DevOps. It can also counteract risks associated with DevOps implementation by structuring this domain. However, the development of such a definition is not possible at this stage of DevOps research, as a stronger focus on practice and how DevOps is defined and understood is necessary.

In conclusion, to our knowledge this is the first publication that provides an overview of DevOps definitions within a DevOps domain context-specific environment. Another outstanding feature of this study is that the individual definitions are based on conceptualized research streams. Thus, a state-of-the-art overview of the topic area, its definitions and main influences can be given. This study

contributes to the literature by providing a unique conceptualization of DevOps definitions with regard to the most influencing factors on DevOps research. In summary, the results of this work confirm the problem of the lack of definition. The various context-specific definitions reflect the needs, application areas and goals of DevOps practice and research and can find direct application.

5.3 Results III: A Supportive DevOps Implementation Decision Framework

In the following, the developed supporting DevOps decision implementation framework is discussed based on its implications and limitations. In general, the framework provides an overview of which line of DevOps definition should be used to implement DevOps with respect to given challenges and corresponding goals of IS development. These challenges and goals are considered to recommend an appropriate DevOps definition and scope for DevOps implementation. Based on the definition problem of DevOps in practice and research, the developed framework supports the adaptation of DevOps definitions to a context-specific application domain.

At first, the framework addresses the respective goals and challenges of modern IS development. These goals and challenges differ in their objectives and impact on an organization. Therefore, they have been divided into different areas of influence on IS development during the development of the framework. This provides a better overview of the driving factors within IS Development. Thus, a practical reference of the framework based on the underlying research question is enabled. Goals and challenges are addressed by the framework. In this way, a better understanding and application of the framework in the organizational setting can be ensured.

Furthermore, the underlying framework is based on a conceptualization of the DevOps topic area. For this purpose, five leading DevOps research streams were identified and conceptualized. Furthermore, the areas were enriched with corresponding DevOps definitions. Based on this, the framework takes a holistic consideration at the DevOps research area. Thus, users of the framework can build on a profound foundation of relevant areas for DevOps. This is supported by the focus on high-quality DevOps literature on which the framework is based. In terms of a practice-oriented use of the framework, it can also provide the necessary viewpoint. This is based on the large number of practice-oriented case studies and surveys, as well

as proceedings and article publications examined. The consideration of guiding topics in the framework allows identification of areas that are relevant to both practice and research in terms of the frameworks' usage. To the best of our knowledge, the framework conceptualizes influential DevOps research areas. Thus, a holistic consideration of the topic area can be provided. Based on the underlying research question, this holistic view promotes DevOps implementation and enables the application to the current DevOps environment.

Third, the framework includes target groups for the five categories. These target groups represent how interventions can be realized when implementing DevOps. Addressing the respective target groups with the corresponding definitions enables achieving the respective goals and solving the given challenges.

In general, the framework combines the three presented areas. The framework links general IS development with DevOps measures and identifies which areas are to be addressed as part of a DevOps implementation. Thereby, a DevOps category can also be considered across its definitions and the initial aim of the framework. Linking to the challenges and goals in IS development enables direct identification of relevant DevOps domains across its definitions for DevOps implementation. Thus, the framework is also intended to address influencing viewpoints of DevOps in IS development. The versatile representation of the framework in terms of DevOps and IS development enables a holistic view of the environment.

The framework includes two underlying limitations. First, the practical relevance of the framework can be considered. The development of the framework is based on high quality DevOps literature that is largely based on practical observations (as seen in *Table 10: Concept Matrix*). However, the practical viewpoint was only considered from a research perspective within the examined publications. Consequently, possible practice-relevant areas are not included in the framework. Furthermore, it must be noted that DevOps is a constantly evolving subject area and the framework will need to be iterated in the future to cover relevant topics.

To our knowledge, this is the first approach that has developed a framework that combines the current IS development environment with DevOps in a context-specific manner. When applied more specifically to DevOps implementation, the framework enables direct application in a practical context. By incorporating goals and challenges, current practical problems and needs can be identified. It thus represents a lightweight applicability. In addition, the framework can help researchers and

organizations get an overview of the topic area to choose the appropriate definition for their approach. Future research can extend the framework to adapt it to changing needs and problems in IS development.

5.3 Further general Implications and Limitations

In this section, a brief description of the further cross-result implications and limitations of this work is provided once again.

The results of this thesis form interconnected results which were created and elaborated on the basis of systematic literature review. In this context, general limitations of these interconnected results can be observed. First of all, the selection of the literature has to be considered. The selection is mainly limited by the availability of the publications. In this regard, a small number of publications that might have been relevant could not be considered. Nevertheless, interlinkages between the different publications could be determined by the forward and backward search. Thereby, connections between missing and included publications are observed which counteracts the availability of certain publications. Moreover, the selected publications give a broad overview of the DevOps research domain.

In addition, the focus of this thesis can be traced back to the systematic literature review and the identified research streams. In the systematic literature review, the individual concepts were identified and structured according to the guidelines of Webster & Watson (2002). This methodology allowed the scientific identification of relevant DevOps research streams. As a result, the five categories were derived from current quality DevOps research and build the base for this thesis' results.

Furthermore, it must be emphasized that DevOps is a strongly changing and still evolving research domain (Beyer et al., 2018, p. 7), which is assumed to continue in the future. Therefore, the results of this thesis and DevOps research in general require further iterations in future research due to the characteristics of the DevOps research domain. However, this thesis' results allow for adaptations to future research streams and developments in DevOps research. Nevertheless, the framework and the other results summarize the current state of DevOps research. This allows adjustments as well as adaptations to changing circumstances.

6 Conclusion

DevOps is a research field with a strong practical focus (Ozkaya, 2019, pp. 4-5; Wiedemann et al., 2019, pp. 48-49). It is characterized by continuous development and continuous new approaches to application (Hüttermann, 2021, pp. 1-4; Krey et al., 2022, p. 7307).

In this thesis, the DevOps research field was reviewed based on a systematic literature review based on Webster & Watson (2002) and Levy & Ellis (2006). Thereby, a concept matrix was elaborated as a first intermediate result of this thesis. Within this scope, five DevOps research streams were identified and conceptualized within DevOps research. Thus, 33 high-quality publications from DevOps research were considered and conceptualized based on covered DevOps topics. As a result, the research streams *DevOps Practices*, *DevOps Principles*, *DevOps Groups*, *DevOps Management*, and *DevOps Tools* were identified within the process of this thesis.

As a second intermediate result, context-specific DevOps definitions were elaborated and derived based on these five research streams and the examined 33 publications. In general, the definitions were extracted from existing DevOps definitions in the considered publications. Thereby, the five identified research streams were used to categorize the DevOps definitions accordingly. Thus, this thesis provides five context-specific DevOps definitions with regard to the most influencing factors on DevOps research.

Based on the first two intermediate results, a supportive decision DevOps implementation framework was developed. The framework connects the context-specific DevOps definitions with goals and challenges of IS development. To address the applicability of the framework, target groups were included into the framework. Regarding DevOps implementation, this enables a specific application area for addressing the underlying framework.

Within the scope of this thesis, various influences on the DevOps domain could be identified. First, DevOps research is an evolving and emerging topic (Beyer et al., 2018, p. 7). However, at its young age of 15 years, it already changed in perception regarding its research streams. While the initial research and practical focus on DevOps was practical and tangible regarding DevOps implementation in practice, it changed to a more profound and top-down perspective. Thereby, an increasing focus on cultural aspects such as management, team structures and responsibilities as well

as the transition to DevOps can be observed in research (as seen in *Table 10: Concept Matrix*). Furthermore, considering the current body of knowledge of DevOps literature, DevOps Practices and DevOps Principles have a predominant focus. In the context of this thesis and its results, it is assumed that the categories DevOps Principles and DevOps Practices will merge with the other identified categories. Thereby, this is assumed to give DevOps a more defined scope and to counteract the general definition problem of DevOps.

Based on the changing characteristics of the DevOps research domain, this indicates a need for continuous iteration of results and the research domain in general in order to cover future relevant DevOps domains and topics. This thesis recommends future research on DevOps to continuous elaboration of the domain and its research streams as it is still evolving and highly changing.

Second, considering the context-specific aspect of DevOps, this thesis elaborated context-specific interconnected results. As DevOps is a continuously evolving domain, a context-specific view of DevOps becomes necessary. This results in the need for DevOps definitions to be tailored to a specific context (Hemon-Hildgen & Rowe, 2022, p. 569; Hüttermann, 2021, pp. 5-11). Within the development of the framework and the intermediate results, the context-specific background of DevOps was considered and applied to fit into the broad research domain. The identified research streams allow context reference for the elaborated definitions and framework. Consequently, this enables a diverse view of the research field and allows the results of this thesis to fit the context-specific requirements of this domain.

Furthermore, the context-specific characteristics of DevOps indicate that the DevOps research field requires multi-dimensional considerations. The developed framework shows a first approach for a multi-dimensional approach which considers various aspects of the DevOps domain. Such an approach can enable interlinkages and trade-offs between different areas that are relevant to be considered in the context of DevOps.

Lastly, considering DevOps in the context of modern IS development, DevOps implementation can help achieve respective goals and tackle given challenges. As DevOps represents a multi-dimensional approach, it impacts diverse areas of an organization. Thereby, focusing collaboration and communication in the implementation process can facilitate processes and remove barriers between various important organizational areas.

As a result, DevOps requires a practical focus within research. It is recommended that future research on DevOps focuses DevOps from a practical viewpoint. Thus, a context-specific approach is recommended which allows a holistic consideration of the domain. This can enable the research area to be conceptualized from a practical perspective and thereby, counteract the current definition and understanding problems in DevOps research and practice. This approach is assumed to conceptualize patterns in definitions, application, and measures. However, DevOps domain including frameworks and definitions require for elaboration to deal with continuously changing domain environment.

In conclusion, based on the scientific results presented, it is evident that a transition towards intentional context-specific DevOps research with high practical relevance will determine future research on DevOps.

Literature

- Akbar, M. A., Rafi, S., Alsanad, A. A., Qadri, S. F., Alsanad, A., & Alothaim, A. (2022). Toward Successful DevOps: A Decision-Making Framework. *IEEE Access*, 10, 51343–51362. <https://doi.org/10.1109/ACCESS.2022.3174094>.
- Alt, R., Auth, G., & Kögler, C. (2017). Innovationsorientiertes IT-Management mit DevOps. In R. Alt, G. Auth, & C. Kögler, *Innovationsorientiertes IT-Management mit DevOps* (S. 21–32). Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-18704-0_3.
- Amaro, R., Pereira, R., & da Silva, M. M. (2023). Capabilities and Practices in DevOps: A Multivocal Literature Review. *IEEE Transactions on Software Engineering*, 49(2), 883–901. <https://doi.org/10.1109/TSE.2022.3166626>.
- Bass, L. (2018). The Software Architect and DevOps. *IEEE Software*, 35(1), 8–10. <https://doi.org/10.1109/MS.2017.4541051>.
- Bass, L., Weber, I. M., & Zhu, L. (2015). *DevOps: A software architect's perspective*. Addison-Wesley Professional.
- Beck, K., Beedle, M., van Bennekum, A. Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J. Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*. Retrieved February 06, 2023, from <https://agilemanifesto.org/>.
- Beetz, F., & Harrer, S. (2022). GitOps: The Evolution of DevOps? *IEEE Software*, 39(4), 70–75. <https://doi.org/10.1109/MS.2021.3119106>.
- Beyer, B., Murphy, N. R., Rensin, D. K., Kawahara, K., & Thorne, S. (Hrsg.). (2018). *The site reliability workbook: Practical ways to implement SRE*. O'Reilly Media.
- Díaz, J., López-Fernández, D., Pérez, J., & González-Prieto, Á. (2021). Why are many businesses instilling a DevOps culture into their organization? *Empirical Software Engineering*, 26(2), 25. <https://doi.org/10.1007/s10664-020-09919-3>.
- Disterer, G. (2011). ITIL-basierte Inbetriebnahme neuer Anwendungen. *HMD Praxis der Wirtschaftsinformatik*, 48(2), 48–57. <https://doi.org/10.1007/BF03340567>.
- Dornenburg, E. (2018). The Path to DevOps. *IEEE Software*, 35(5), 71–75. <https://doi.org/10.1109/MS.2018.290110337>.

- Dyck, A., Penners, R., & Lichter, H. (2015). Towards Definitions for Release Engineering and DevOps. *2015 IEEE/ACM 3rd International Workshop on Release Engineering*, 3–3. <https://doi.org/10.1109/RELENG.2015.10>.
- Ebert, C., Gallardo, G., Hernantes, J., & Serrano, N. (2016). DevOps. *IEEE Software*, 33(3), 94–100. <https://doi.org/10.1109/MS.2016.68>.
- Elsevier Inc., (2023). *AIS eLibrary*. . Retrieved February 20, 2023, from <https://aisel.aisnet.org/>.
- Erich, F. M. A., Amrit, C., & Daneva, M. (2017). A qualitative study of DevOps usage in practice. *Journal of Software: Evolution and Process*, 29(6). <https://doi.org/10.1002/smr.1885>.
- Faustino, J., Adriano, D., Amaro, R., Pereira, R., & da Silva, M. M. (2022). DEVOPS benefits: A systematic literature review. *Software: Practice and Experience*, 52(9), 1905–1926. <https://doi.org/10.1002/spe.3096>.
- Gall, M. & Pigni, F. (2018). Leveraging DevOps for mission critical software. *AMCIS 2018 Proceedings*. 2. <https://aisel.aisnet.org/amcis2018/ITProjMgmt/Presentations/2>.
- Galup, S., Dattero, R., & Quan, J. (2020). What do agile, lean, and ITIL mean to DevOps? *Communications of the ACM*, 63(10), 48–53. <https://doi.org/10.1145/3372114>.
- Ghantous, G. B., & Gill, A. (2017). DevOps: Concepts, Practices, Tools, Benefits and Challenges. *PACIS 2017 Proceedings*. 96. <https://aisel.aisnet.org/pacis2017/96>.
- Hemon, A., Fitzgerald, B., Lyonnet, B., & Rowe, F. (2020). Innovative Practices for Knowledge Sharing in Large-Scale DevOps. *IEEE Software*, 37(3), 30–37. <https://doi.org/10.1109/MS.2019.2958900>.
- Hemon, A., Lyonnet, B., Rowe, F., & Fitzgerald, B. (2019). Conceptualizing the Transition from Agile to DevOps: A Maturity Model for a Smarter IS Function. In A. Elbanna, Y. K. Dwivedi, D. Bunker, & D. Wastell (Hrsg.), *Smart Working, Living and Organising* (Bd. 533, S. 209–223). Springer International Publishing. https://doi.org/10.1007/978-3-030-04315-5_15.
- Hemon-Hildgen, A., & Rowe, F. (2022). Conceptualising and defining DevOps: A review for understanding, not a framework for practitioners. *European Journal of Information Systems*, 31(5), 568–574. <https://doi.org/10.1080/0960085X.2022.2100061>.

- Hemon-Hildgen, A., Rowe, F., & Monnier-Senicourt, L. (2020). Orchestrating automation and sharing in DevOps teams: A revelatory case of job satisfaction factors, risk and work conditions. *European Journal of Information Systems*, 29(5), 474–499. <https://doi.org/10.1080/0960085X.2020.1782276>.
- Highsmith, J. A. (2010). *Agile project management: Creating innovative products* (2nd ed). Addison-Wesley.
- Humble, J., & Farley, D. (2010). *Continuous delivery: Reliable software releases through build, test, and deployment automation*. Addison-Wesley.
- Humble, J., & Molesky, J. (2011). Why enterprises must adopt devops to enable continuous delivery. *Cutter IT Journal*, 24(8), 6.
- Hüttermann, M. (2012). *DevOps for developers*. Apress, New York.
- Hüttermann, M. (2021). The DevOps Continuum: Walking the Shadowy Bridge from Information Systems Development to Operations. ECIS 2021 Research Papers. 78. https://aisel.aisnet.org/ecis2021_rp/78.
- Hüttermann, M., & Rosenkranz, C. (2019). Devops: Walking the shadowy bridge from development success to information systems success. ICIS 2019 Proceedings. 10. https://aisel.aisnet.org/icis2019/is_development/is_development/10
- Jabbari, R., bin Ali, N., Petersen, K., & Tanveer, B. (2016). What is DevOps?: A Systematic Mapping Study on Definitions and Practices. *Proceedings of the Scientific Workshop Proceedings of XP2016*, 1–11. <https://doi.org/10.1145/2962695.2962707>.
- Kersten, M. (2018). Mining the Ground Truth of Enterprise Toolchains. *IEEE Software*, 35(3), 12–17. <https://doi.org/10.1109/MS.2018.2141029>
- Krey, M., Kabbout, A., Osmani, L., & Saliji, A. (2022). DevOps adoption: challenges & barriers. In 55th Hawaii International Conference on System Sciences (HICSS), virtual, 3-7 January 2022 (pp. 7297-7309). University of Hawai'i at Manoa.
- Leite, L., Rocha, C., Kon, F., Milojevic, D., & Meirelles, P. (2020). A Survey of DevOps Concepts and Challenges. *ACM Computing Surveys*, 52(6), 1–35. <https://doi.org/10.1145/3359981>.
- Leshem, S., & Trafford, V. (2007). Overlooking the conceptual framework. *Innovations in Education and Teaching International*, 44(1), 93–105. <https://doi.org/10.1080/14703290601081407>.

- Levy, Y., & J. Ellis, T. (2006). A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research. *Informing Science: The International Journal of an Emerging Transdiscipline*, 9, 181–212. <https://doi.org/10.28945/479>.
- López-Fernandez, D., Diaz, J., Garcia, J., Perez, J., & Gonzalez-Prieto, A. (2022). DevOps Team Structures: Characterization and Implications. *IEEE Transactions on Software Engineering*, 48(10), 3716–3736. <https://doi.org/10.1109/TSE.2021.3102982>.
- Lwakatare, L. E., Karvonen, T., Sauvola, T., Kuvaja, P., Olsson, H. H., Bosch, J., & Oivo, M. (2016a). Towards DevOps in the embedded systems domain: Why is it so hard?. In 2016 49th hawaii international conference on system sciences (hicss) (pp. 5437-5446). IEEE.
- Lwakatare, L.E., Kuvaja, P. and Oivo, M. (2016b), “An exploratory study of DevOps: Extending the dimensions of devops with practices”, ICSEA 2016 : The Eleventh International Conference on Software Engineering Advances, No. August, pp. 91–99.
- Macdonald, E. K., Wilson, H., Martinez, V., & Toossi, A. (2011). Assessing value-in-use: A conceptual framework and exploratory study. *Industrial Marketing Management*, 40(5), 671–682. <https://doi.org/10.1016/j.indmarman.2011.05.006>.
- Marnewick, C., & Langerman, J. (2021). DevOps and Organizational Performance: The Fallacy of Chasing Maturity. *IEEE Software*, 38(5), 48–55. <https://doi.org/10.1109/MS.2020.3023298>.
- Okumus, F. (2001), "Towards a strategy implementation framework", *International Journal of Contemporary Hospitality Management*, Vol. 13 No. 7, pp. 327-338. <https://doi.org/10.1108/09596110110403712>.
- Ouhbi, S., & Pombo, N. (2020). Software Engineering Education: Challenges and Perspectives. *2020 IEEE Global Engineering Education Conference (EDUCON)*, 202–209. <https://doi.org/10.1109/EDUCON45650.2020.9125353>
- Ozkaya, I. (2019). Are DevOps and Automation Our Next Silver Bullet? *IEEE Software*, 36(4), 3–95. <https://doi.org/10.1109/MS.2019.2910943>.
- Ozkaya, I. (2020). The Deployment View. *IEEE Software*, 37(3), 3–5. <https://doi.org/10.1109/MS.2020.2971573>.

- Perez, J. E., Gonzalez-Prieto, A., Diaz, J., Lopez-Fernandez, D., Garcia-Martin, J., & Yague, A. (2022). DevOps Research-Based Teaching Using Qualitative Research and Inter-Coder Agreement. *IEEE Transactions on Software Engineering*, 48(9), 3378–3393. <https://doi.org/10.1109/TSE.2021.3092705>.
- Rafi, S., Akbar, M. A., Mahmood, S., Alsanad, A., & Alothaim, A. (2022). Selection of DevOps best test practices: A hybrid approach using ISM and fuzzy TOPSIS analysis. *Journal of Software: Evolution and Process*, 34(5). <https://doi.org/10.1002/smr.2448>.
- Rowse, M., & Cohen, J. (2021, January). A survey of DevOps in the south African software context. In Proceedings of the 54th Hawaii International Conference on System Sciences (p. 6785 - 6794).
- Sharp, J., & Babb, J. (2018). Is Information Systems late to the party? The current state of DevOps research in the Association for Information Systems eLibrary. AMCIS 2018 Proceedings. 26. <https://aisel.aisnet.org/amcis2018/AdvancesIS/Presentations/26>.
- Shropshire, J., Menard, P., and Sweeney, B. (2017) Uncertainty, Personality, and Attitudes toward DevOps. AMCIS 2017 Proceedings. 28. <https://aisel.aisnet.org/amcis2017/AdoptionIT/Presentations/28>.
- Silva, M., Faustino, J., Pereira, R., & Mira Da Silva, M. (2018). Productivity Gains of DevOps Adoption in an IT Team: A Case Study. In B. Andersson, B. Johansson, S. Carlsson, C. Barry, M. Lang, H. Linger, & C. Schneider (Eds.), *Designing Digitalization (ISD2018 Proceedings)*. Lund, Sweden: Lund University. ISBN: 978-91-7753-876-9. <http://aisel.aisnet.org/isd2014/proceedings2018/ISDevelopment/8>.
- Smart, J. (2018). To Transform to Have Agility, Dont Do a Capital A, Capital T Agile Transformation. *IEEE Software*, 35(6), 56–60. <https://doi.org/10.1109/MS.2018.4321245>.
- Sommerville, I. (2020a). *Engineering software products: An introduction to modern software engineering* (First edition). Pearson.
- Sommerville, I. (2020b). *Modernes Software-Engineering: Entwurf und Entwicklung von Softwareprodukten*. Pearson.
- Stoica, M., Mircea, M., & Ghilic-Micu, B. (2013). Software Development: Agile vs. Traditional. *Informatica Economica*, 17(4/2013), 64–76. <https://doi.org/10.12948/issn14531305/17.4.2013.06>.

- Stray, V., Moe, N. B., & Aasheim, A. (2019). Dependency management in large-scale agile: a case study of DevOps teams. In Proceeding of the 52nd Hawaii International Conference on System Sciences (HICSS 2019). University of Hawai'i.
- Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS quarterly*, 26(2), xiii-xxiii. <https://www.jstor.org/stable/4132319>.
- Wiedemann, A. (2018). IT governance mechanisms for DevOps Teams-How incumbent companies achieve competitive advantages. In Proceedings of the 51st hawaii international conference on system sciences.
- Wiedemann, A., Forsgren, N., Wiese, M., Gewalt, H., & Krüger, H. (2019). Research for practice: The DevOps phenomenon. *Communications of the ACM*, 62(8), 44–49. <https://doi.org/10.1145/3331138>.
- Wiedemann, A., Wiese, M., Gewalt, H., & Krüger, H. (2020). Understanding how DevOps aligns development and operations: A tripartite model of intra-IT alignment. *European Journal of Information Systems*, 29(5), 458–473. <https://doi.org/10.1080/0960085X.2020.1782277>.
- Wolff, E. (2016). *Continuous Delivery: Der pragmatische Einstieg* (2., aktualisierte und erweiterte Auflage). dpunkt.verlag.
- Zhu, L., Bass, L., & Champlin-Scharff, G. (2016). DevOps and Its Practices. *IEEE Software*, 33(3), 32–34. <https://doi.org/10.1109/MS.2016.81>.

A. Appendix: Journal List and Proceedings Publication (Levis & Ellis, 2006)

Note: From “A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research”, by Y. Levy & T. J. Ellis, 2006, *Informing Science: The International Journal of an Emerging Transdiscipline*, 9, pp. 186-187. <https://doi.org/10.28945/479>

Ranked MIS Journals (Based on ISWorld)		Literature Vendor (Database)											
		ProQuest (ABI/INFORM)	Elsevier (ScienceDirect)	INFORMS	IEEE (Comp Soc & Xplore)	Wilson (OmniFile)	Thomson (G. Bus, OneFile)	ACM (Digital Lib)	JSTOR	Blackwell (Synergy)	LEA Journals	EBSCOhost	Full Text Web Access
No.	Journal Name												
1	MIS Quarterly	✓	-	-	-	✓	✓	-	✓	-	-	-	-
2	Information Systems Research	✓	-	✓	-	-	✓	-	-	-	-	✓	-
3	Communications of the ACM	✓	-	-	-	-	✓	✓	-	-	-	-	-
4	Management Science	✓	-	✓	-	-	✓	-	✓	-	-	✓	-
5	Journal of MIS	✓	-	-	-	✓	-	-	-	-	-	-	-
6	Artificial Intelligence	-	✓	-	-	-	-	-	-	-	-	-	-
7	Decision Sciences	✓	-	-	-	-	-	-	-	✓	-	-	-
8	Harvard Business Review	-	-	-	-	-	✓	-	-	-	-	-	-
9	IEEE Transactions (various)	-	-	-	✓	-	✓	-	-	-	-	-	-
10	AI Magazine	✓	-	-	-	✓	✓	-	-	-	-	-	-
11	European Journal of IS	✓	-	-	-	-	-	-	-	-	-	-	-
12	Decision Support Systems	-	✓	-	-	-	-	-	-	-	-	-	-
13	IEEE Software	-	-	-	✓	-	-	-	-	-	-	-	✓
14	Information & Management	-	✓	-	-	-	-	-	-	-	-	-	-
15	ACM Transactions on DB Sys	-	-	-	-	-	✓	✓	-	-	-	-	-
16	IEEE Trans on Software Eng	-	-	-	✓	-	-	-	-	-	-	-	-
17	ACM Transactions (various)	✓	-	-	-	-	✓	✓	-	-	-	-	-
18	J of Computer and System Sci	-	✓	-	-	-	-	-	-	-	-	-	-
19	Sloan Management Review	✓	-	-	-	✓	✓	-	-	-	-	-	-
20	Communications of the AIS	-	-	-	-	-	-	✓	-	-	-	✓	-
21	IEEE Trans on Sys, Man, & Cyb	-	-	-	✓	-	-	-	-	-	-	-	-
22	ACM Computing Surveys	✓	-	-	-	-	✓	✓	-	-	-	-	-
23	Journal on Computing	✓	-	-	-	-	✓	-	-	-	-	✓	-
24	Academy of Management Journal	✓	-	-	-	-	✓	-	✓	-	-	-	-
25	Int'l J of Electronic Commerce	-	-	-	-	✓	-	-	-	-	-	-	-
26	Journal of the AIS	-	-	-	-	-	-	✓	-	-	-	-	-
27	IEEE Transactions on Computers	-	-	-	✓	-	-	-	-	-	-	-	-
28	Information Systems Frontiers	✓	-	-	-	-	-	-	-	-	-	-	-
29	Journal of Management Systems	✓	-	-	-	✓	-	-	-	-	-	-	-
30	Organization Science	✓	-	✓	-	-	✓	-	✓	-	-	✓	-
31	IEEE Computer	-	-	-	✓	-	-	-	-	-	-	-	-
32	Information Systems Journal	-	-	-	-	-	-	-	-	✓	-	-	-
33	Administrative Science Quarterly	✓	-	-	-	✓	✓	-	✓	-	-	-	-
34	J of Global Info Management	✓	-	-	-	-	✓	-	-	-	-	-	-
35	The DB for Advances in IS	✓	-	-	-	-	-	-	-	-	-	-	-
36	Journal of Database Management	✓	-	-	-	-	✓	-	-	-	-	-	-
37	Information Systems	-	✓	-	-	-	-	-	-	-	-	-	-
38	Academy of Management Review	✓	-	-	-	-	✓	-	✓	-	-	-	-
39	Journal of the ACM	✓	-	-	-	-	✓	✓	-	-	-	-	-
40	Computers & Operations Research	-	✓	-	-	-	-	-	-	-	-	-	-
41	Human-Computer Interaction	-	-	-	-	-	-	-	-	-	✓	-	-
42	California Management Review	✓	-	-	-	-	-	-	-	-	-	-	-
43	Information Technology & People	✓	-	-	-	-	-	-	-	-	-	-	-
44	Journal of Strategic IS	-	✓	-	-	-	-	-	-	-	-	-	-
45	Journal of Global IT Management	✓	-	-	-	-	-	-	-	-	-	-	-
46	ACM Transactions on IS	-	-	-	-	-	✓	✓	-	-	-	-	-
47	Informing Science	-	-	-	-	-	-	-	-	-	-	✓	✓
48	Journal of Information Management	✓	-	-	-	-	-	-	-	-	-	-	-
49	Operations Research	✓	-	-	-	-	✓	-	✓	-	-	✓	-
50	Journal of Computer IS	✓	-	-	-	-	-	-	-	-	-	-	-

Ranked and Non-Ranked IS Conferences Ranked Order is Based on Hardgrave and Walstorm (1997)		Literature Vendor (Database)						
		Elsevier (ScienceDirect)	INFORMS	IEEE (Comp Soc & Xplore)	ACM (Digital Lib)	Full Text Web Access (Fee)	Full Text Web Access (Free)	Proc. CD-ROM for purchase
No.	Conference Name							
1	International Conference on Information Systems (ICIS)	-	-	-	✓	-	-	-
2	Hawaii International Conference on System Sciences (HICSS)	-	-	✓	-	-	✓	-
3	International Federation for Information Processing (IFIP)	-	-	-	✓	✓	-	✓
4	International Conference on Decision Support Systems (DSS)	-	-	-	✓	-	-	-
5	Decision Sciences Institute (DSI) - National Conference	-	-	-	-	✓	-	-
6	Society of Information Management (SIM) Conference	✓	-	-	-	-	-	-
7	International Association for Computer Information Systems (IACIS) Conference (Proceedings published in Issues in Information Systems)	-	-	-	-	✓	-	✓
8	Institute for Operations Research and the Management Sciences (INFORMS) Conference	-	✓	-	-	-	-	-
9	Information Resources Management Association (IRMA) Conference	-	-	-	-	-	-	✓
10	Academy of Management (AOM) Conference	-	-	-	-	✓	-	-
11	Decision Sciences Institute (DSI) - Regional Conferences	-	-	-	-	✓	-	✓
NR	International Academy of Information Management (IAIM) Conference	-	-	-	-	✓	-	✓
NR	American Conference on Information Systems (AMCIS)	-	-	-	✓	-	-	-
NR	Information Systems Education Conference (ISECON)	-	-	-	-	-	✓	✓
NR	Institute of Electrical and Electronics Engineers (IEEE) National Conferences	-	-	✓	-	-	-	-
NR	Informing Science + Information Technology Education (InSITE) Conference	-	-	-	-	-	✓	✓

No. - Indicates the rank of a conference in general by Hardgrave and Walstorm (1997)'s study

NR - Non-ranked

B. Appendix: Overview of all Databases

Name of the database: ProQuest

Date of search: 10.10.2022

Procedure	Hits
Keywords	5252
Language	5181
Journal List	40
Removing redundant papers	39
Relevance: abstract and title	23
Full paper accessibility	22
Relevance: full paper	13
Amount of forward search paper	1
Amount of backward search paper	5
Sum Long List	13
Sum Short List	19

Name of the database: EBSCOhost

Date of search: 11.10.2022

Procedure	Hits
Keywords	4066
Language	3018
Journal List	26
Removing redundant papers	26
Relevance: abstract and title	12
Full paper accessibility	10
Relevance: full paper	5
Amount of forward search paper	0
Amount of backward search paper	0
Sum Long List	5
Sum Short List	0

Name of the database: Elsevier (Science Direct)**Date of search: 11.10.2022**

Procedure	Hits
Keywords	698
Language	698
Journal List	5
Removing redundant papers	5
Relevance: abstract and title	0
Full paper accessibility	0
Relevance: full paper	0
Amount of forward search paper	0
Amount of backward search paper	0
Sum Long List	0
Sum Short List	0

Name of the database: IEEE Comp Soc & Xplore**Date of search: 17.10.2022**

Procedure	Hits
Keywords	1727
Language	1727
Journal List	41
Removing redundant papers	39
Relevance: abstract and title	2
Full paper accessibility	2
Relevance: full paper	2
Amount of forward search paper	0
Amount of backward search paper	0
Sum Long List	2

Sum Short List	2
-----------------------	----------

Name of the database: ACM Digital Library

Date of search: 11.10.2022

Procedure	Hits
Keywords	1306
Language	1306
Journal List	14
Removing redundant papers	14
Relevance: abstract and title	1
Full paper accessibility	1
Relevance: full paper	0
Amount of forward search paper	0
Amount of backward search paper	0
Sum Long List	1
Sum Short List	1

Name of the database: AIS AISNET

Date of search: 04.11.2022

Procedure	Hits
Keywords	47
Language	46
Journal List	36
Removing redundant papers	36
Relevance: abstract and title	12
Full paper accessibility	9
Relevance: full paper	9
Amount of forward search paper	1
Amount of backward search paper	2
Sum Long List	9
Sum Short List	3

Total of all Databases

Procedure	Hits
Keywords	13096
Language	11976
Journal List	162
Relevance: abstract and title	50
Full paper accessibility	44
Removing redundant papers	33
Relevance: full paper	26

Amount of forward search paper (without duplicates)	2
Amount of backward search paper (without duplicates)	5
Sum Final List	33

C. Appendix: Timely Overview of Publications

Row Labels	Count of Publications
2016	4
2017	2
2018	9
2019	6
2020	4
2021	4
2022	4
Grand Total	33

D. Appendix: DevOps Definitions Overview

Definition	Source	Original	DevOps Category
[DevOps as] a set of principles and practices that promote greater communication and cooperation among key stakeholders for the goals of defining, creating, and operating systems and software products or services, as well as continual improvement in all elements of that entity's life cycle.	Amaro et al., 2023	Olszewska et al., 2021	(1) DevOps Principles (2) DevOps Practices
DevOps is a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality	(1) Bass, 2018 (2) Ozkaya, 2019 (3) Zhu et al., 2016	Bass et al., 2015	(1) DevOps Practices

The term DevOps is a portmanteau of the words Development and Operations (1). [...] DevOps expresses the idea of improving the collaboration between Development and Operations teams in corporations or even joining these two teams together. [...] (2).	Beetz & Harrer, 2022	(1) Davis & Daniels, 2016 (2) Sato, 2014	(1) DevOps Principles (2) DevOps Groups
DevOps integrates the two worlds of development and operations, using automated development, deployment, and infrastructure monitoring. It's an organizational shift in which, instead of distributed siloed groups performing functions separately, cross-functional teams work on continuous operational feature deliveries.	Ebert et al., 2016	Own Definition	(1) DevOps Management
DevOps comprises the development and operation of information systems (1). DevOps claims to enable shorter response times when it comes to the delivery of features or bug fixes utilizing a continuous integration and continuous deployment (CI/CD) pipeline (2).	Gall & Pigni, 2018	(1) Ghantous & Gill, 2017 (2) Humble & Farley, 2010	(1) DevOps Practices (2) DevOps Tools
Delivering value to the business through software requires processes and coordination that often span multiple teams across complex systems and involves developing and delivering software with both quality and resiliency.	Galup et al., 2020	Forsgren & Kersten, 2018	(1) DevOps Management

DevOps is a set of practices that is trying to bridge developer-operations gap at the core of things and at the same time covers all the aspects which help in speedy, optimized and high quality software delivery	Ghantous & Gill, 2017	Nagpal & Shadab, 2014	(1) DevOps Practices
[...] DevOps strives for a continuous delivery of value through continuous integration, delivery and deployment. DevOps is hence an extension of agile to the entire software delivery pipeline, aiming to optimize lead time between code writing and its use by end-users in a real production environment.	Hemon et al., 2020	Own Definition	(1) DevOps Practices (2) DevOps Tools
DevOps philosophy leads to build bridges between Development and Operations teams. DevOps foster the creation of cross-functional teams where each team member need to consider and anticipate the job to be done by other members.	Hemon et al., 2019	Own Definition	(1) DevOps Principles (2) DevOps Groups
DevOps as set of a principles for collaborative work implemented between the IS design and development function and the IS operations function, along with potentially other constituencies (stakeholders), which is founded on the sharing of culture, goals, measures, automation tools and automated processes towards continuous delivery of software.	Hemon-Hildgen et al., 2020	Own Definition	(1) DevOps Principles

DevOps is a portmanteau word of “development” and “operations”. [...] “IT development” (those that build IS) and “IT operations” (those that run and maintain IS) increasingly apply shared goals and use shared practices across both functions, bringing together team members from both development and operations, in order to implement information system development (ISD) in a comprehensive way.	Hüttermann, 2021	(1) Lwakatare, Kuvaja, et al., 2016 (2) Qumer Gill et al., 2018 (3) Wiedemann, 2017	(1) DevOps Principles
DevOps is a cultural movement used to facilitate rapid software development.	Krey et al., 2022	Macarthy & Bass, 2020	
DevOps is a collaborative and multidisciplinary effort within an organization to automate continuous delivery of new software versions, while guaranteeing their correctness and reliability.	Leite et al., 2020	Dyck et al., 2015	(1) DevOps Practices
DevOps can be defined as a cultural movement to improve and accelerate the delivery of business value by making the collaboration between development and operations effective.	López-Fernández et al., 2022	Own Definition	(1) DevOps Principles (2) DevOps Management
DevOps is a new phenomenon in software engineering, emphasizing collaboration, automation, virtualization and new tools that bridge software development and operations activities [1]. A blend of	Lwakatare et al., 2016a	(1) Humble & Farley, 2010 (2) Own Definition	(1) DevOps Principles (2) DevOps Practices

the words ‘development’ and ‘operations’, DevOps constitutes both technical and non-technical practices that help software-intensive companies to increase responsiveness to customer needs through frequent and automated software releases.			
DevOps– a blend of two words Development and Operations– is about aligning incentives of everybody involved in delivering software, with particular emphasis on developers, testers and operations personnel.	Lwakatare et al., 2016b	(1) Humble & Molesky, 2011 (2) Tesse, & Iden, 2008	(1) DevOps Groups
[...] [DevOps is] a development methodology aimed at bridging the gap between Development (Dev) and Operations, emphasizing communication and collaboration, continuous integration, quality assurance and delivery with automated deployment utilizing a set of development practices	(1) Marnewick & Langerman, 2021 (2) Sharp & Babb, 2018	Jabbari et al., 2016	(1) DevOps Principles (2) DevOps Practices
DevOps (Development & Operations) is an emerging paradigm for software development and operations (1,2) [...] DevOps is both a culture shift in software delivery (3), and a set of practices for continuous integration, delivery and deployment (CI/CD) supported by technology enablers (4).	Rowse & Cohen, 2021	(1) Fitzgerald & Stol, 2017 (2) Jabbari et al., 2016 (3) Walls, 2013 (4) Senapathi et al., 2018	(1) DevOps Principles (2) DevOps Practices

[...] [DevOps is] a conceptual framework which aims at benefitting IS development by integrating development and operations in various ways	Sharp & Babb, 2018	Erich et al., 2014	(1) DevOps Principles
DevOps is a fusion of development and operations functions among the various information technology groups within organizations. [...] (1) DevOps is not software, a set of tools, or a job description. It is a set of policies, practices, and values that streamline the software implementation process (2). [...] DevOps is conceptually defined in terms of five dimensions (3). These dimensions include the following: collaboration, automation, culture, monitoring, and measurement.	Shorpshire et al., 2017	(1) Dyck et al., 2015 (2) Roche, 2013 (3) Lwakatare et al., 2016b	(1) DevOps Principles (2) DevOps Practices
[...] [DevOps is] a conceptual framework that is supported on a culture of collaboration, automation, measurement, and information sharing.	Silva et al., 2018	Erich et al., 2014	(1) DevOps Principles
DevOps is about rapid, flexible development iterations through domain-crossing team compositions that break complex architecture and features sets into small chunks that can be produced and deployed independently.	Stray et al., 2019	Ebert et al., 2016	(1) DevOps Groups

DevOps is a clipped compound of the words development and operation. The process it refers to entails strong collaboration between development and operations, automation, and use of new tools and technologies.	Wiedemann, 2018	(1) Humble & Farley, 2011 (2) Lwakatare et al., 2016a	(1) DevOps Principles
DevOps combines activities of software development and delivery to enhance the speed of getting new software features to customers. [...] [DevOps] is summarized as CALMS: culture, automation, lean, measurement, and sharing.	Wiedemann et al., 2019	(1) Forsgren & Humble, 2015 (2) Humble & Molesky, 2011	(1) DevOps Principles (2) DevOps Practices
The DevOps method integrates the tasks, knowledge and skills pertaining to planning, building, and running software product activities in a joint cross-functional team within the IT function.	Wiedemann et al., 2020	Wiedemann et al., 2019	(1) DevOps Principles (2) DevOps Groups

Eidesstaatliche Versicherung