University of Stuttgart

Hochschule Reutlingen
Reutlingen University

Towards an
Architecture-centric Methodology for Migrating to Microservices

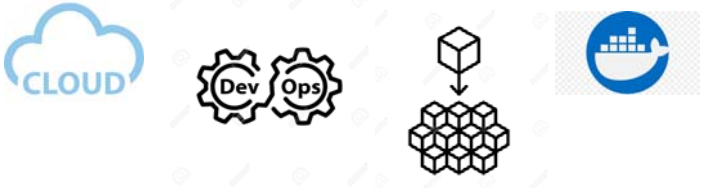
Jonas Fritzsch, Justus Bogner, Markus Haug,
Stefan Wagner, Alfred Zimmermann

13/06/2022

University of Stuttgart / Reutlingen University of Applied Sciences, Germany

Application Modernization: Migrating to Microservices

New architectural pattern / paradigm for app. development



Companies struggle to migrate their existing applications

On the public forum, tech leaders proclaim: “Just move to the cloud, or you won’t be competitive”. Then it turns out the migration process is shrouded in complexity as no realistic answers are easily available online.

Marek Gajda
CTO, The Software House
https://tsh.io/blog/cloud-migration/

Research Objective and Questions

Design, implement and evaluate
A practically applicable methodology
For migrating monolithic applications toward a Microservices architecture

RQ1: What are **intentions**, **strategies** and **challenges** in the context of migrating existing systems to Microservices?

RQ2: What **architectural refactoring techniques** are applicable in the context of decomposing a system into Microservices?

RQ3: What are relevant **quality attributes** and **metrics** for evaluating the appropriateness of service partitioning and service granularity?

RQ4: How can a practically applicable **migration methodology** guide architects?

Research Questions

RQ1: What are **intentions**, **strategies** and **challenges** in the context of migrating existing systems to Microservices?

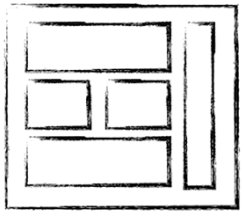
RQ2: What **architectural refactoring techniques** are applicable in the context of decomposing a system into Microservices?

RQ3: What are relevant **quality attributes** and **metrics** for evaluating the appropriateness of service partitioning and service granularity?

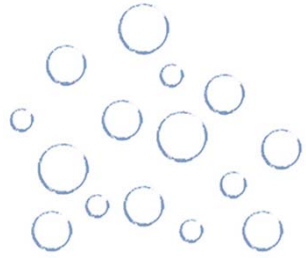
RQ4: How can a practically applicable **migration methodology** guide architects?

RQ1: Intentions, Strategies, Challenges

Interview Study with 16 Practitioners from 10 German-based Companies, 14 Systems [3]



Monolithic Legacy Application



Microservices

Intentions [3]	Strategies [3]	Challenges [3]	Results [2,3]
Maintainability	**Rewrite**	**Decomposition**	▪ Subset of Microservices characteristics implemented
Scalability	**Strangler Pattern**	**Lack of Expertise**	▪ Tendency for fewer and more coarse-grained services
Funct. Requirements	**Extension**	**DevOps and Automation**	▪ DevOps and automation still in an early stage
Operability	**Greenfield**	**Integration**	▪ Wrong service cuts in many cases
Company Strategy	**Continuous Evolution**	**Legacy System**	
Time to Market		**Security**	

[2] J. Bogner, J. Fritzsch, S. Wagner, A. Zimmermann, "Microservices in Industry: Insights into Technologies, Characteristics, and Software Quality.", in IEEE International Conference on Software Architecture Workshops (ICSA-W) IEEE Computer Society, Hamburg, Germany, **2019**

[3] J. Fritzsch, J. Bogner, S. Wagner, A. Zimmermann, "Microservices Migration in Industry: Intentions, Strategies, and Challenges", in 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), Cleveland (Ohio), USA, **2019**

Research Questions

RQ1: What are **intentions**, **strategies** and **challenges** in the context of migrating existing systems to Microservices?

RQ2: What architectural **refactoring techniques** are applicable in the context of decomposing a system into Microservices?

RQ3: What are relevant **quality attributes** and **metrics** for evaluating the appropriateness of service partitioning and service granularity?

RQ4: How can a practically applicable **migration methodology** guide architects?

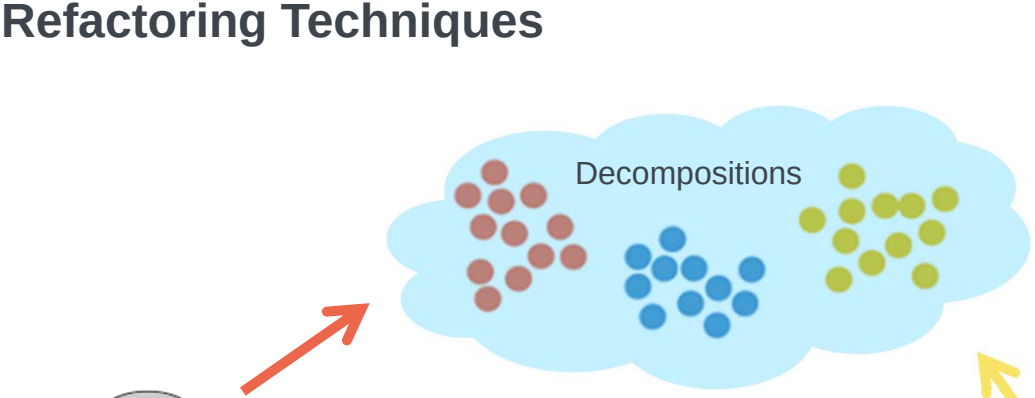
RQ2: Refactoring Techniques

Literature Review and Classification of **10** Refactoring Approaches [1]

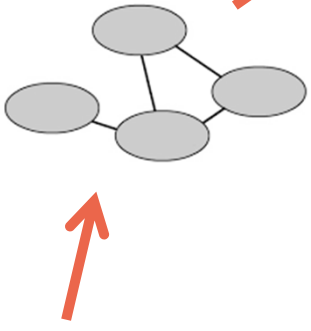
#	Approach	Authors (Year)
1	Towards the understanding and evolution of monolithic applications as microservices	Escobar, et. al. (2016)
2	Towards a Technique for Extracting Microservices from Monolithic Enterprise Systems	Levcovitz, et. al. (2016)
3	Requirements reconciliation for scalable and secure microservice (de)composition	Ahmadvand, et. al. (2016)
4	Microservices Identification Through Interface Analysis	Baresi, et. al. (2017)
5	Service Cutter: A systematic approach to service decomposition	Gysel, et. al. (2016)
6	Extraction of Microservices from Monolithic Software Architectures	Mazlami, et. al. (2017)
7	GranMicro: A Black-Box Based Approach for Optimizing Microservices Based Applications	Mustafa, et. al. (2017)
8	Microservice Ambients: An Architectural Meta-Modelling Approach for Microservice Granularity	Hassan, et. al. (2017)
9	Workload-based Clustering of Coherent Feature Sets in Microservice Architectures	Klock, et. al. (2017)
10	Towards a MicroServices Architecture for Clouds	Procaccianti, et. al. (2016)

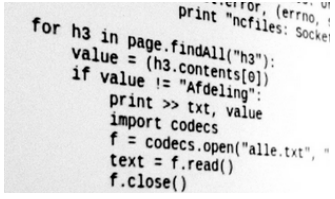
[1] J. Fritzsch, J. Bogner, A. Zimmermann, S. Wagner, "From Monolith to Microservices: A Classification of Refactoring Approaches", in Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment. Cham: Springer International Publishing, **2019**, pp. 128–141.

RQ2: Refactoring Techniques

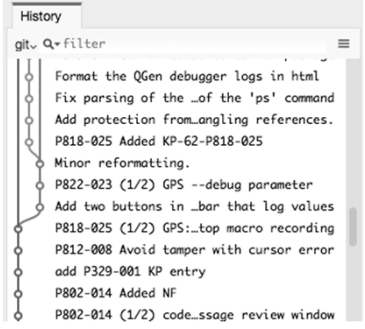


Aggregation

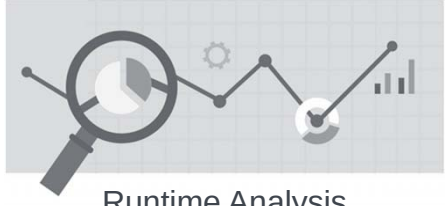




Static Code Analysis



Meta-Data / Model-Driven



Runtime Analysis

Limitations of Existing Migration/Refactoring Approaches

- 10 Refactoring/Migration Approaches reviewed in 2018 [1]
- 31 Refactoring/Migration Approaches reviewed in 2020
  - focus on different requirements and quality attributes
  - applicability limited to certain technologies, languages, architectures (e.g. MVC-Pattern, Java-based or web applications)
  - based of different techniques (see classification in [1])
  - no or only experimental tool support
  - evaluation often insufficient
  - **not considered** by practitioners, or unknown to them [3]

Research Questions

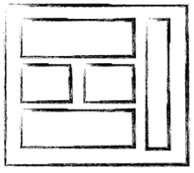
RQ1: What are **intentions**, **strategies** and **challenges** in the context of migrating existing systems to Microservices?

RQ2: What **architectural refactoring techniques** are applicable in the context of decomposing a system into Microservices?

RQ3: What are relevant **quality attributes** and **metrics** for evaluating the appropriateness of service partitioning and service granularity?

RQ4: How can a practically applicable **migration methodology** guide architects?

Quality Attributes

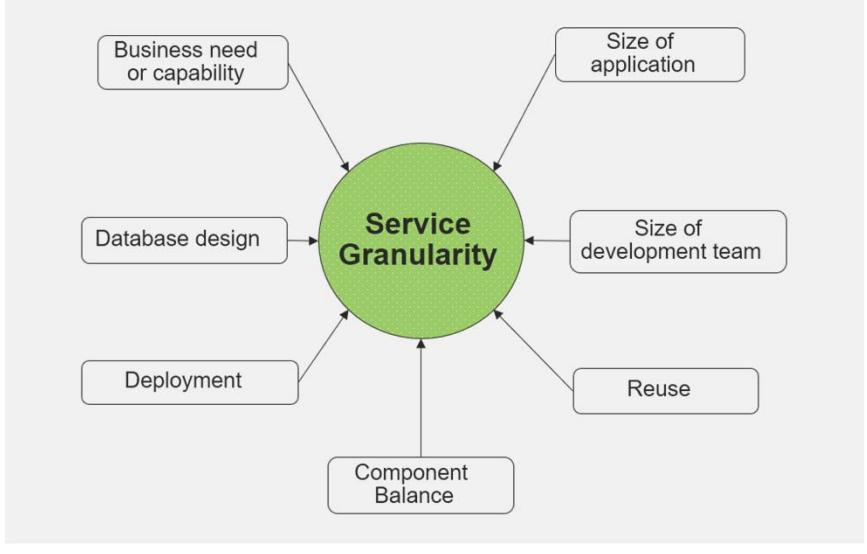


Monolithic Legacy Application

Microservices

Intentions [3]

Maintainability
Scalability
Funct. Requirements
Operability
Company Strategy
Time to Market



[3] J. Fritzsch, J. Bogner, S. Wagner, A. Zimmermann, "Microservices Migration in Industry: Intentions, Strategies, and Challenges", in 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), Cleveland (Ohio), USA, **2019**

Research Questions

RQ1: What are **intentions**, **strategies** and **challenges** in the context of migrating existing systems to Microservices?

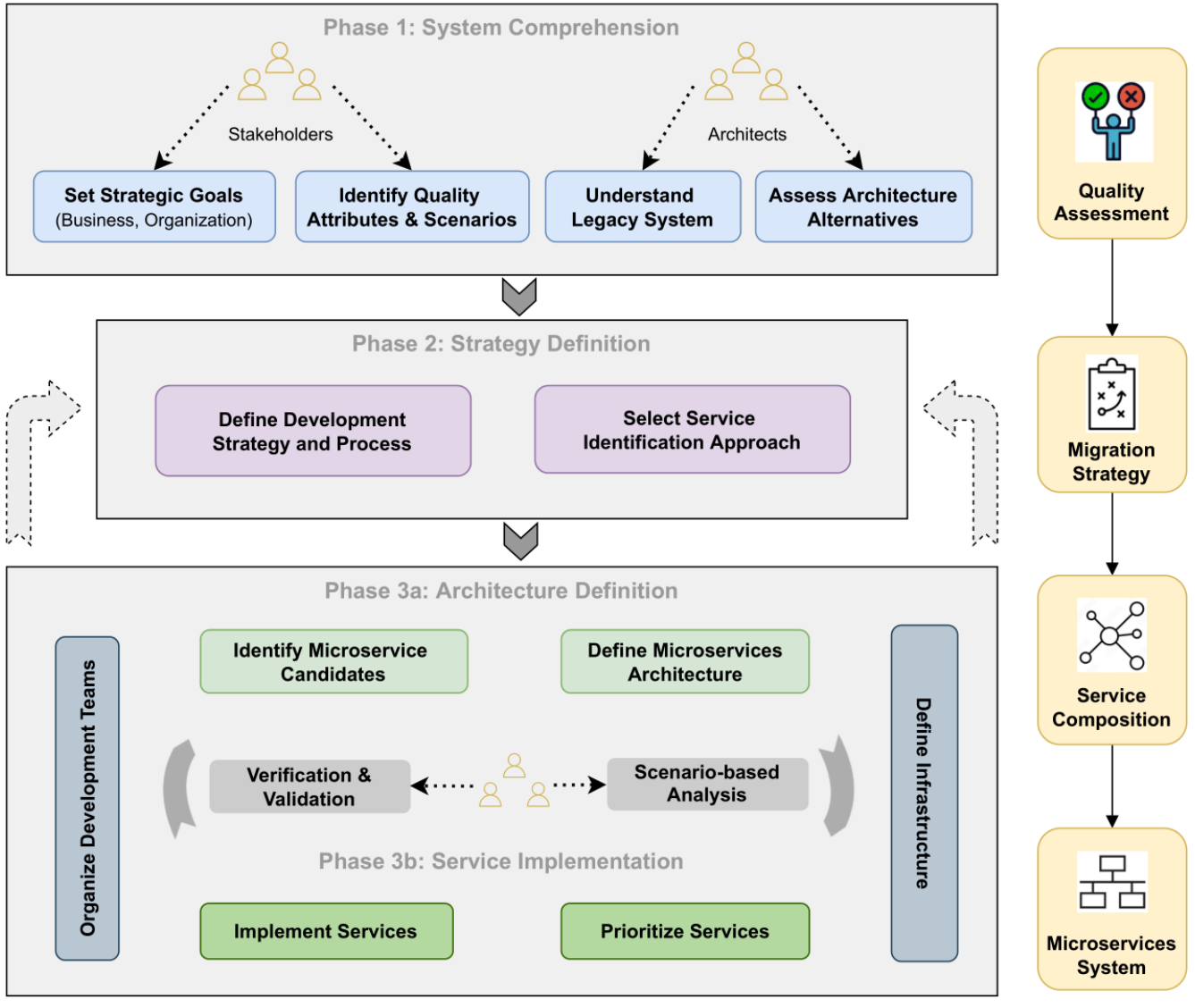
RQ2: What **architectural refactoring techniques** are applicable in the context of decomposing a system into Microservices?

RQ3: What are relevant **quality attributes** and **metrics** for evaluating the appropriateness of service partitioning and service granularity?

RQ4: How can a practically applicable **migration methodology** guide architects?

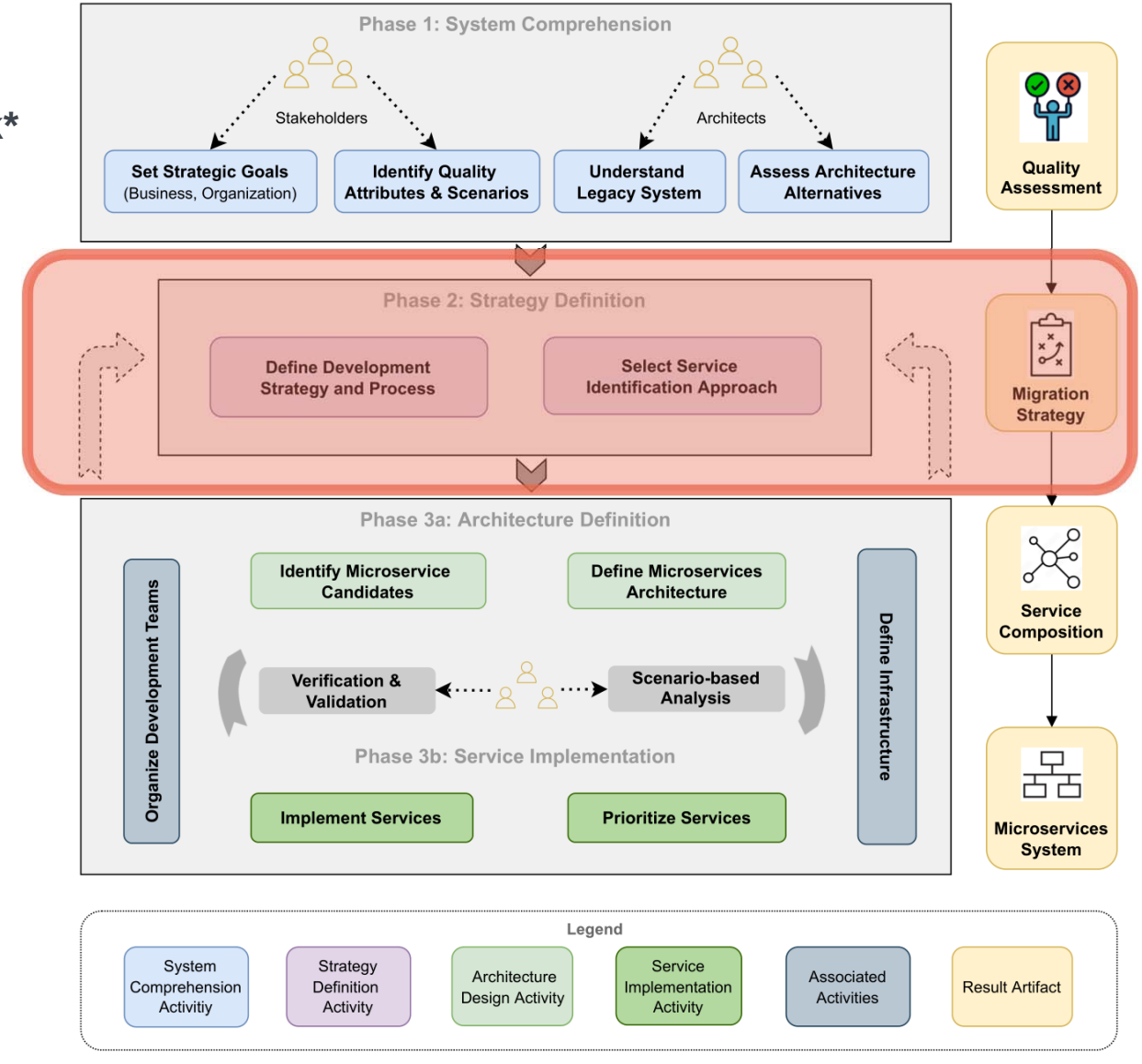
Proposed Framework*

*inspired by works from
Wolfart et al. [4,5]
Taibi et al. [6]
Bozan et al. [7]



Proposed Framework*

*inspired by works from
Wolfart et al. [4,5]
Taibi et al. [6]
Bozan et al. [7]

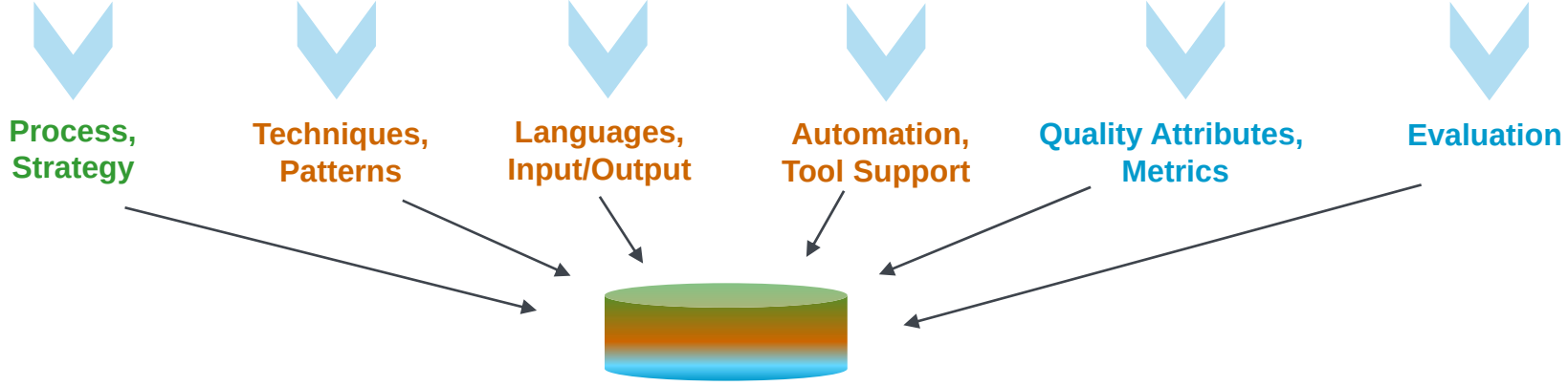


A Structured Repository

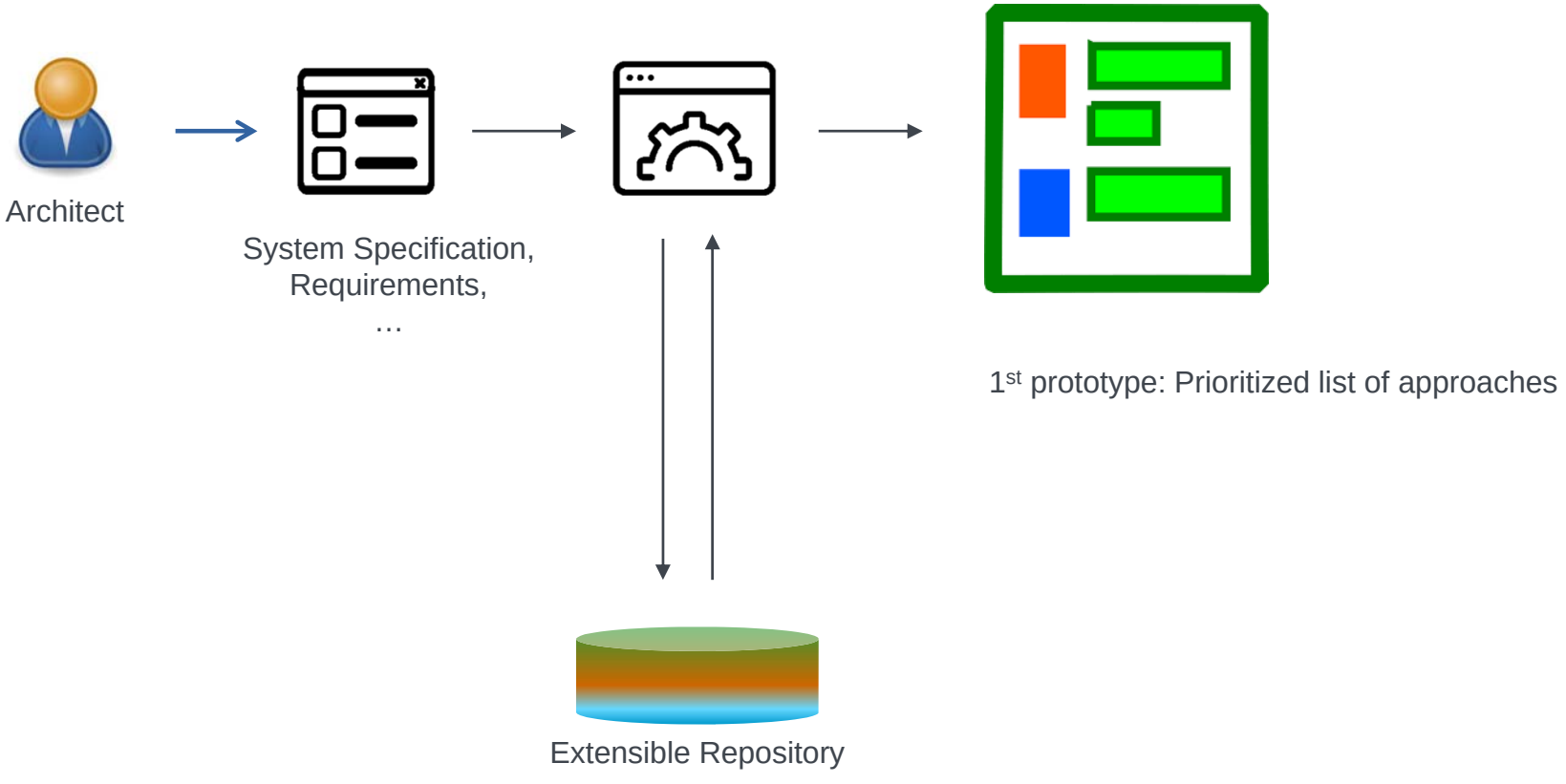
41 Approaches (2020)

#	Approach	Authors (Year)
1	Towards the understanding and evolution of monolithic applications as microservices	Escobar, et. al. (2016)
2	Towards a Technique for Extracting Microservices from Monolithic Enterprise Systems	Levcovitz, et. al. (2016)
3	Requirements reconciliation for scalable and secure microservice (de)composition	Ahmadvand, et. al. (2016)
4	Microservices Identification Through Interface Analysis	Baresi, et. al. (2017)
5	Service Cutter: A systematic approach to service decomposition	Gysel, et. al. (2016)
6	Extraction of Microservices from Monolithic Software Architectures	Mazlami, et. al. (2017)
7	GranMicro: A Black-Box Based Approach for Optimizing Microservices Based Applications	Mustafa, et. al. (2017)
8	Microservice Ambients: An Architectural Meta-Modelling Approach for Microservice Granularity	Hassan, et. al. (2017)
9	Workload-based Clustering of Coherent Feature Sets in Microservice Architectures	Klock, et. al. (2017)
10	Towards a MicroServices Architecture for Clouds	Procaccianti, et. al. (2016)

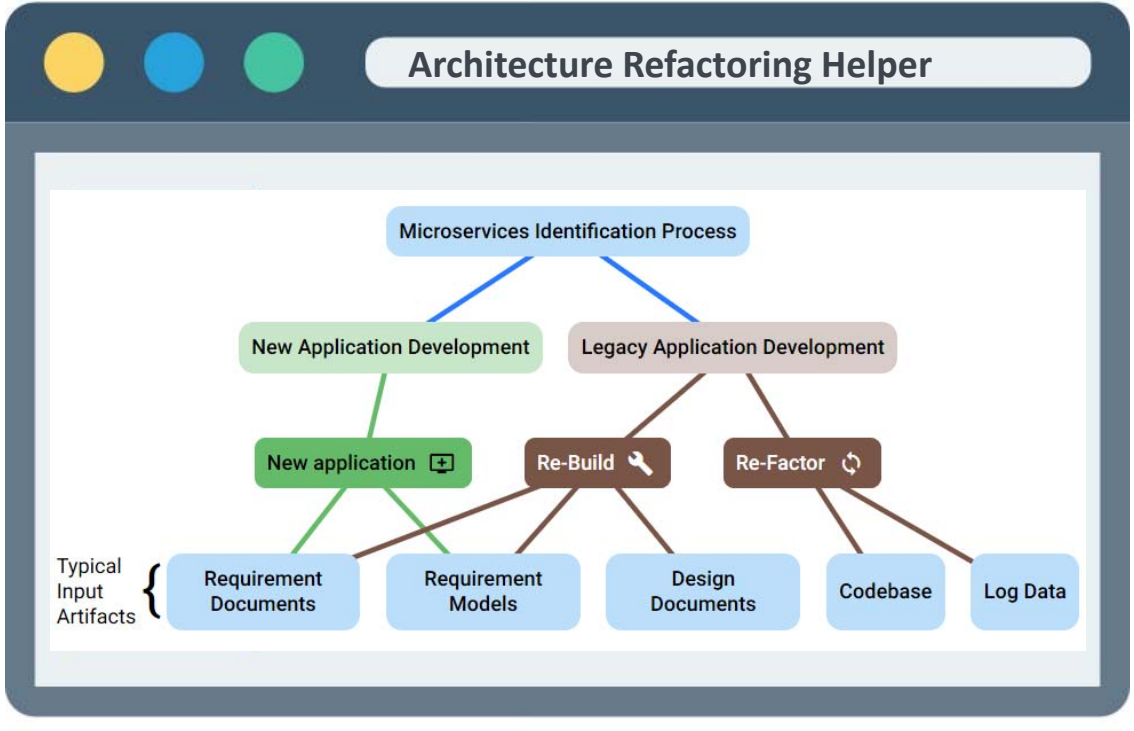
A Probabilistic Approach For Obtaining An Optimized Number Of Services Using Weighted Matrix And Multidimensional Scaling	MD
An Automatic Extraction Approach - Transition to Microservices Architecture from Monolithic Application	SA
Discovering Microservices in Enterprise Systems Using a Business Object Containment Heuristic	MD+SA+DA
Extraction of Microservices from Monolithic Software Architectures	SA
From Monolith to Microservices: A Dataflow-Driven Approach	MD
From Monolithic Systems to Microservices: A Decomposition Framework based on Process Mining	DA
Functionality-oriented Microservice Extraction Based on Execution Trace Clustering	DA
Identifying Microservices Using Functional Decomposition	MD
Microservices Identification Through Interface Analysis	SA
Migrating Monolithic Mobile Application to Microservice Architecture: An Experiment Report	MD
Migrating to Cloud-Native Architectures Using Microservices: An Experience Report	MD
Migrating Web Applications from Monolithic Structure to Microservices Architecture	SA+DA
Object-aware Identification of Microservices	MD
Re-architecting OO Software into Microservices A Quality-Centred Approach	SA
Requirements Reconciliation for Scalable and Secure Microservice (De)composition	MD
Service Cutter: A Systematic Approach to Service Decomposition	MD
Towards a Technique for Extracting Microservices from Monolithic Enterprise Systems	MD
Towards the Understanding and Evolution of Monolithic Applications as Microservices	SA
Unsupervised learning approach for web application auto-decomposition into microservices	SA
Using Microservices for Legacy Software Modernization	MD+SA



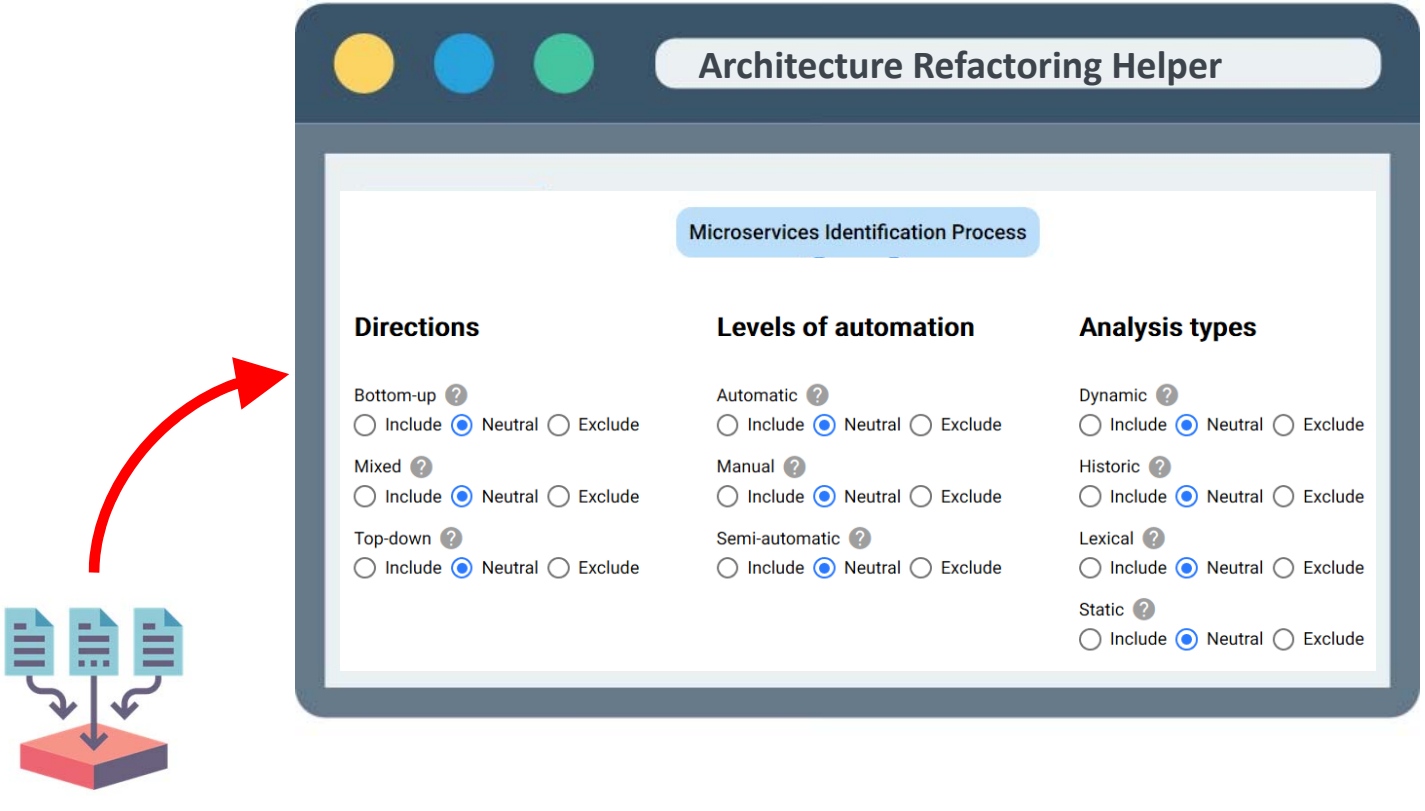
A Meta-Approach to Guide Architectural Refactoring



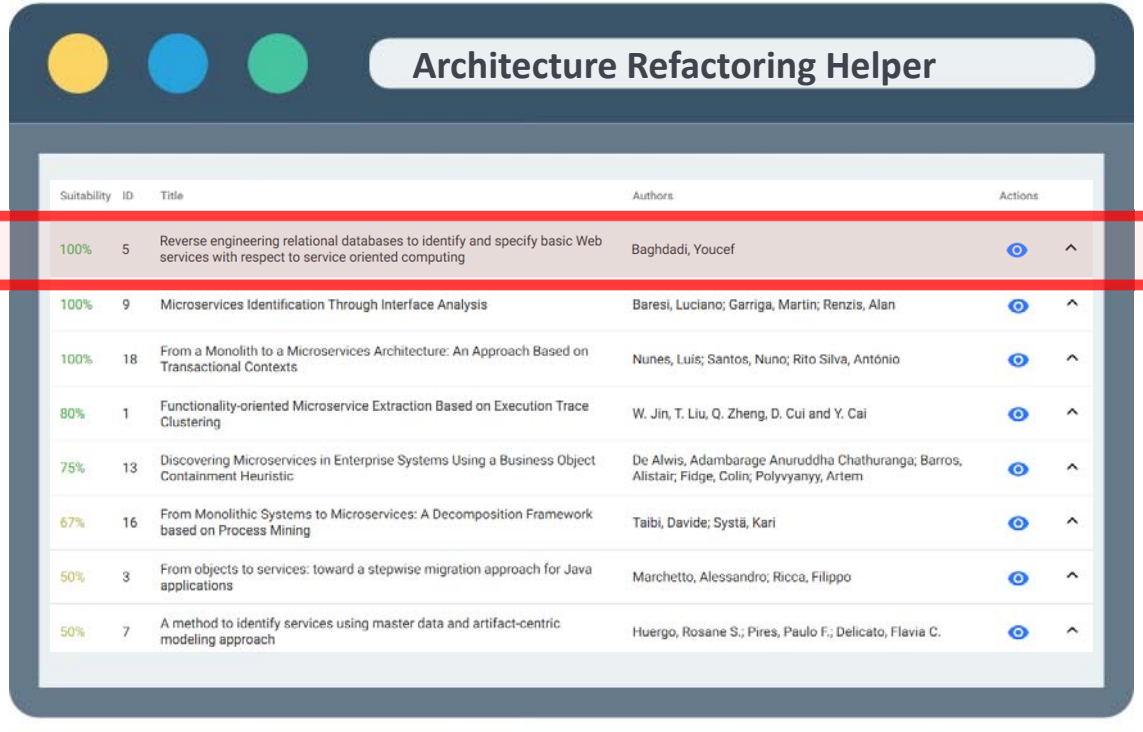
Tool Support (1)



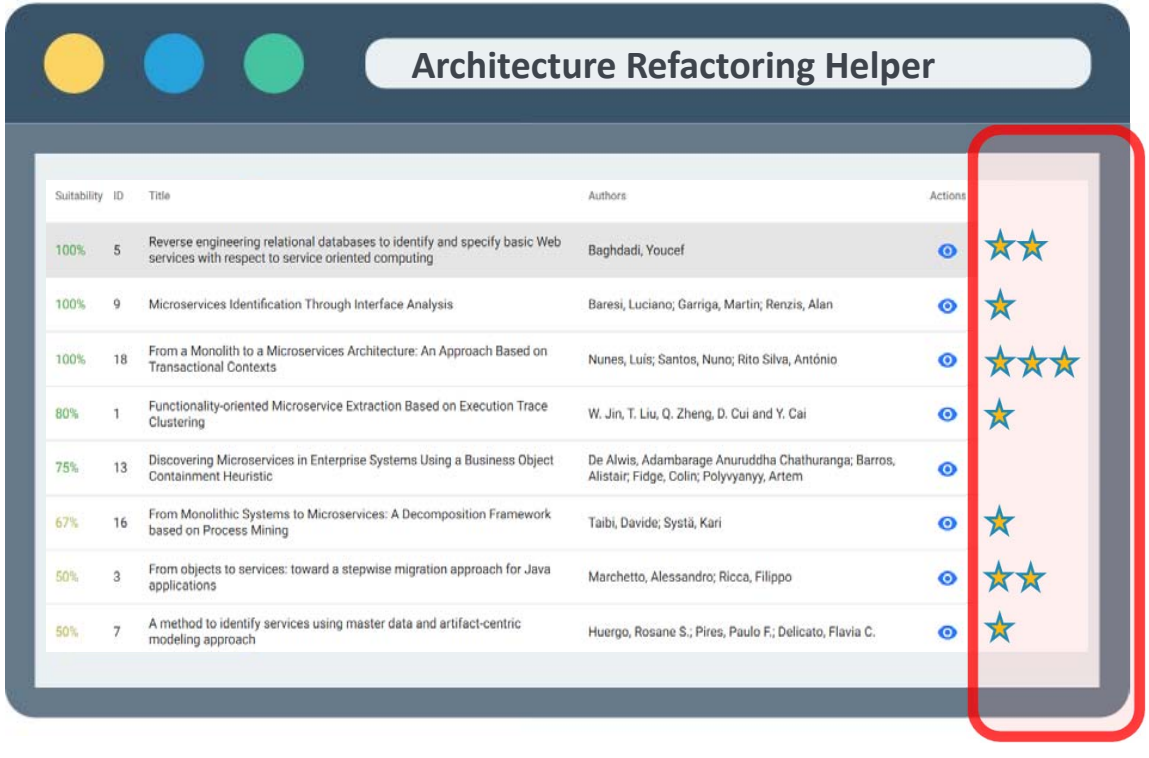
Tool Support (2)



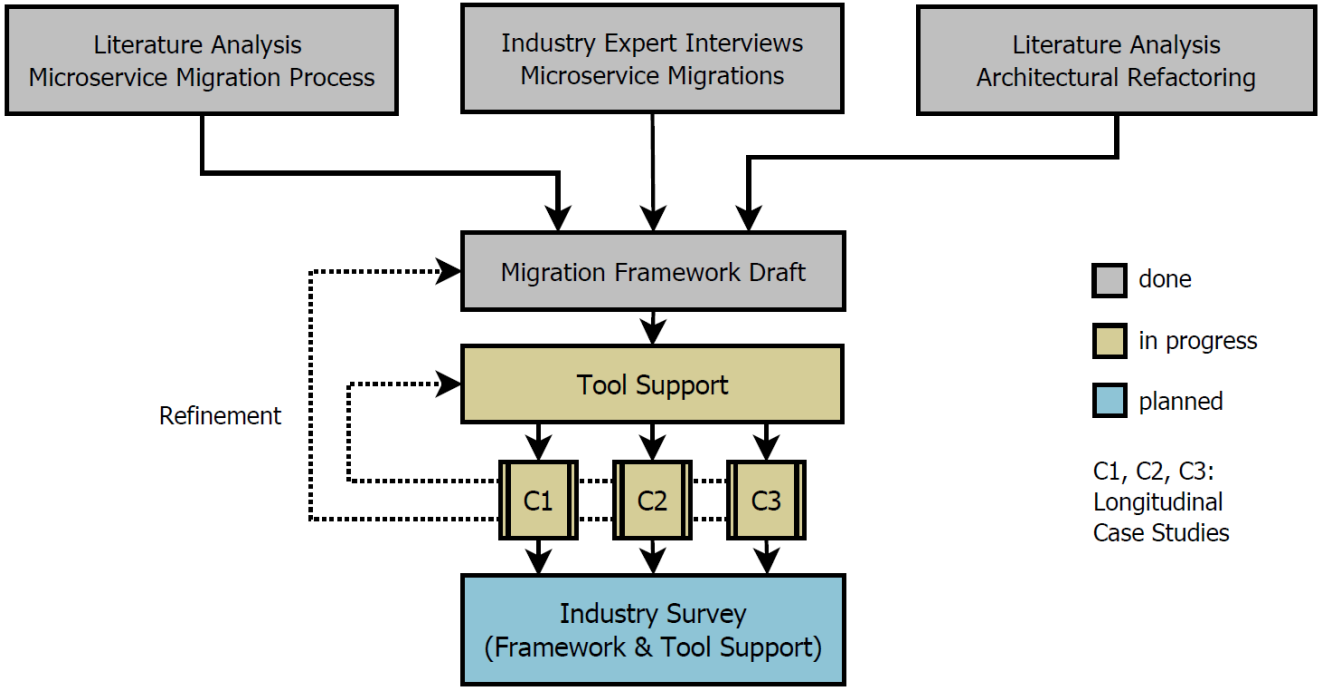
Tool Support (3)



Tool Support (4)



Research Method



References

[1] J. Fritzsch, J. Bogner, A. Zimmermann, S. Wagner, "From Monolith to Microservices: A Classification of Refactoring Approaches", in Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment. Cham: Springer International Publishing, 2019, pp. 128–141.

[2] J. Bogner, J. Fritzsch, S. Wagner, A. Zimmermann, "Microservices in Industry: Insights into Technologies, Characteristics, and Software Quality.", in IEEE International Conference on Software Architecture Workshops (ICSA-W) IEEE Computer Society, Hamburg, Germany, 2019

[3] J. Fritzsch, J. Bogner, S. Wagner, A. Zimmermann, "Microservices Migration in Industry: Intentions, Strategies, and Challenges", in 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME), Cleveland (Ohio), USA, 2019

[4] D. Wolfart et al., "Towards a Process for Migrating Legacy Systems into Microservice Architectural Style," in Anais da IV Escola Regional de Engenharia de Software (ERES 2020), 2020, pp. 255–264

[5] D. Wolfart et al., "Modernizing Legacy Systems with Microservices: A Roadmap," in Evaluation and Assessment in Software Engineering, 2021, pp. 149–159.

[6] D. Taibi, V. Lenarduzzi, and C. Pahl, "Processes, Motivations, and Issues for Migrating to Microservices Architectures: An Empirical Investigation," IEEE Cloud Comput., vol. 4, no. 5, pp. 22–32, Sep. 2017.

[7] K. Bozan, K. Lyytinen, and G. M. Rose, "How to transition incrementally to microservice architecture," Commun. ACM, vol. 64, no. 1, pp. 79–85, Jan. 2021.

University of Stuttgart
Institute of Software Technology

Thank you.

Jonas Fritzsch

e-mail jonas.fritzsch@iste.uni-stuttgart.de
phone +49 (0) 711 685-88458
www.uni-stuttgart.de

University of Stuttgart
Institute of Software Technology
Empirical Software Engineering Group