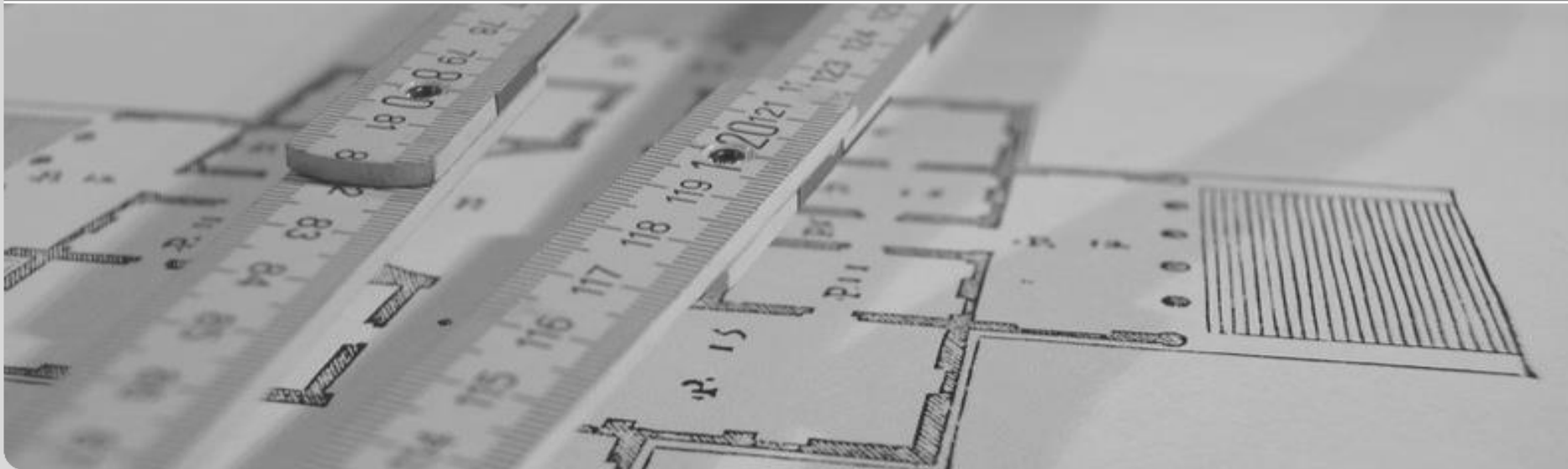


Proposal:

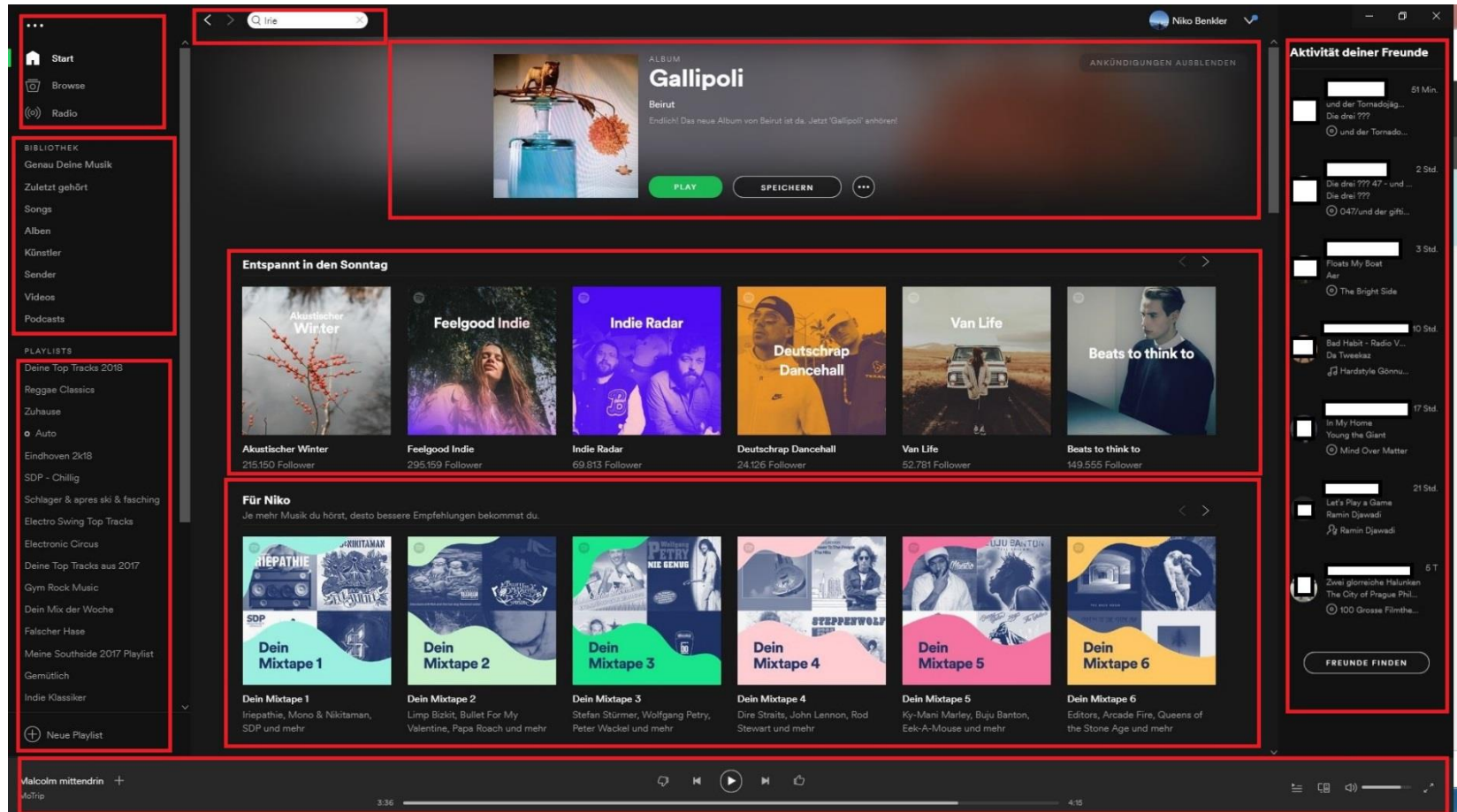
An approach for identifying microservices using clustering on control flow and data flow

Niko Benkler

SOFTWARE DESIGN AND QUALITY GROUP
INSTITUTE FOR PROGRAM STRUCTURES AND DATA ORGANIZATION, FACULTY OF INFORMATICS



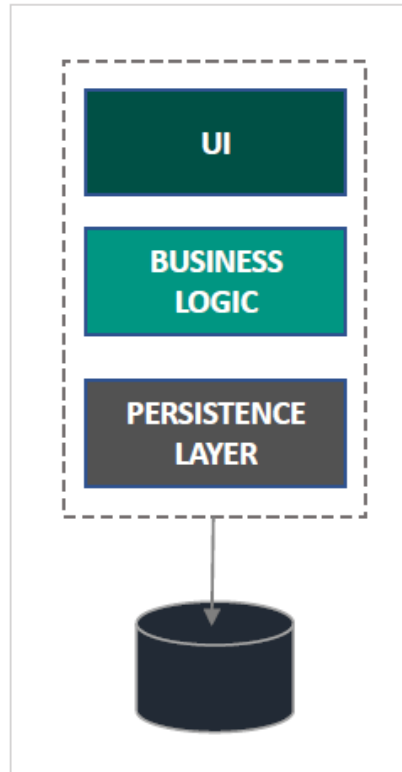
Introductory Example: Spotify



Source: Screenshot, Account of author

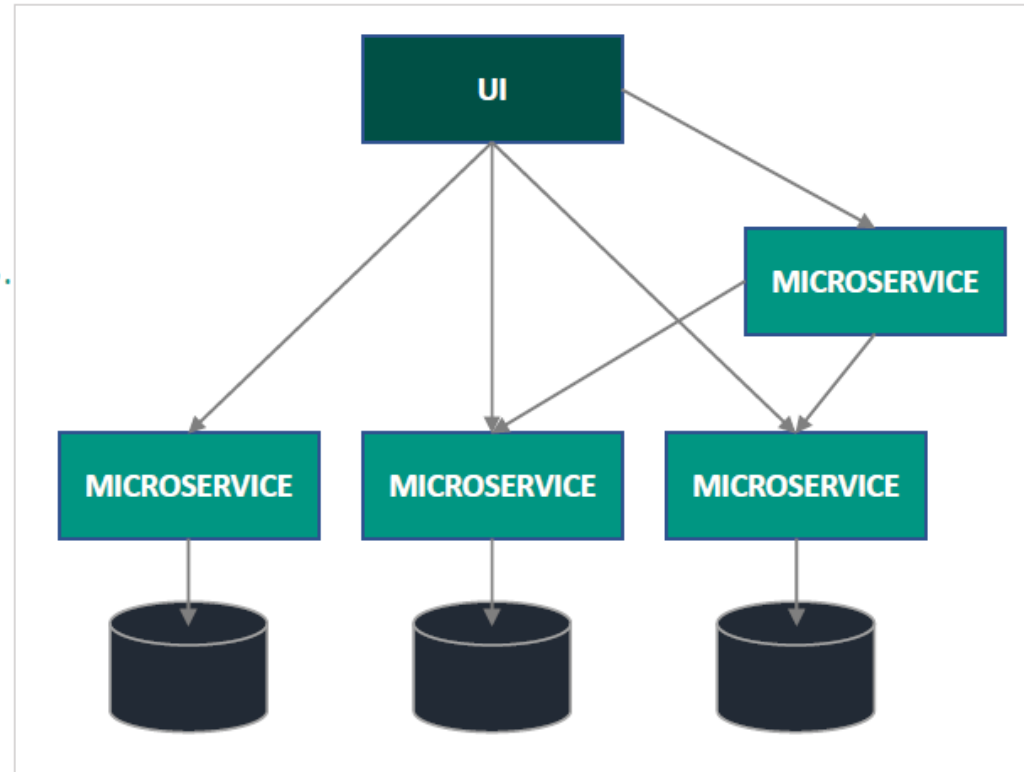
The Architecture - Monolith vs. Microservices

MONOLITHIC



VS.

MICROSERVICE ARCHITECTURE



Microservices: Benefits and Challenges

"The microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API."

- Martin Fowler

Benefits

- Independent Deployment
- Resilience & Fault Isolation
- Improved Scalability
- Neutral development Technology

Challenges

- Expensive Communication
- Infrastructure Automation
- Organizational Restructuring
- Data Consistency
- Microservice Identification

PIBA

Problem

- Migrate a monolithic architecture to a microservice-based system

Idea

- Identify microservices using clustering on control flow and data flow

Benefit

- Faster identification process
- Reduce required expertise and manual effort
- Create more adequate microservices

Action

- Identify possible structural and data object dependencies
- Create separate sets of clusters by using the dependencies as weights
- Match set of clusters to generate possible microservice candidates

Microservice Identification ... so far!

- Complex and highly manual task
- Needs experienced system architects and domain experts
 - ➡ Mainly intuitively and based on experience/knowledge

Risks:

- Too fine grained: Inefficient due to expensive inter-service calls
- Too coarse grained: Loss of scalability and Independence
 - ➡ Formal approach to identify microservices is required

Research Questions

- **RQ1: Which is the most appropriate strategy to extract microservices from a monolithic system?**
 - Literature research, compare based on criteria

- **RQ2: What formal approach can be constructed to perform the extraction process without detailed know-how and manual effort?**
 - Use most adequate strategy from RQ1 as basis for new approach
 - Elaborate new approach

- **RQ3: What is the quality of the microservices recommended by the approach?**
 - Apply to CoCoME and compare with manual decomposition and paper

Approaches I

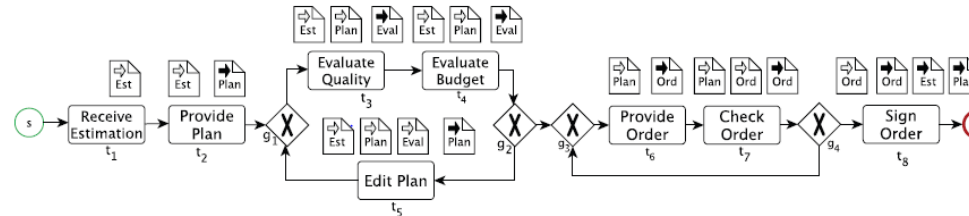
Approach/Criterion	Munezero et al. [22]	Chen et al. [5]	Alwis et al. [6]	Gysel et al. [11]
Basic Concept	define business capabilities by using domain-driven design patterns	algorithmic identification of microservices using data flows	graph-based identification process using heuristics to describe call graph similarities	service decomposition based on 16 coupling criteria
Prerequisites	domain defined by ubiquitous language	systems's data flows constructen on users' natural language description	Log files of legacy system	various System Specification Artifacts (SSAs) in specified format
Input	well defined domain model	Data Flow Diagrams (DFD)	Call Graphs, Source Code, System Database	instances of SSAs (e.g. ERM models, use cases)
Tool support	n/a	n/a	External tool for generating call graphs	implementation and wiki available
Degree of human involvement	domain experts define boundaries for business responsibilities	manual construction of purified DFD	no interaction needed	priorization of coupling criteria
Granularity	depends on the size of the defined business capability	most fine-grained ms candidates in terms of data operations	lowest granularity of sw based on structural and behavioural properties	n/a
Validation	demonstrated on sample domain	two case studies verified against relevant microservice principles and results of [11]	two experiemtns with complex enterprise systems (legacy vs. ms implementation)	validation via implementation and two case studies
Limitation	only conceptional approach, requires vast amount of expertise	transforming purified DFD not trivial (identifying same data operations requires expertise)	requires expressive log files to generate call graphs and identify business object relationships	generating SSAs in specified format is work intense

Approaches II

Approach/Criterion	Mazlami et al. [20]	Amiri [2]	Baresi et al. [3]	Tyszberowicz et al. [24]
Basic Concept	meta-data aided graph clustering	business process oriented graph clustering	semantic similarity of OpenApi specification	functional decomposition of sw requirements
Prerequisites	applications with meaningful VCS data	business processes and entities available	well-defined Api with proper naming	specification of software requirements
Input	Source Code and VCS meta data	BPMN business processes with data object reads and writes	reference vocabulary (fitness function), OpenApi specifications	use cases
Tool support	prototype available (https://github.com/gmaz/frontend)	Clustering tool "Bunch"	experimental prototype (https://github.com/mgariga/decomposer)	use external graph visualize and analyse tools
Degree of human involvement	choose amount of clusters that will represent the microservices	no interaction needed	user defines level of hierarchy	manual elimination of synonyms, irrelevant nouns and verbs
Granularity	depends on chosen amount of clusters	depends on iteration of genetic algorithm for convergence of fitness function	depends on chosen hierarchy level, varies from one to many	depends on size of business capability
Validation	experiments using open-source projects with VCS data (200 to 25000 commits, 1000 to 500000 LOC, 5 to 200 authors)	multiple experiments, results compared with domain experts knowledge	452 OpenApi specification, 5 samples compared with results of sw-engineers and [11]	case study, compared to three manual implementations
Limitation	need meaningful VCS data and ORM model for its data entities	given weight definitions lack formal explanation	depends on reference vocabulary and well-defined interfaces	manual revision of operations (nouns) and state variable (verbs)

Basic Approach: *Object-aware Identification of Microservices*

BPMN Graph:



Structural Dependency

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
t_1	0	1	0	0	0	0	0	0
t_2	0	0	1	0	0	0	0	0
t_3	0	0	0	1	0	0	0	0
t_4	0	0	0	0	1	1	0	0
t_5	0	0	1	0	0	0	0	0
t_6	0	0	0	0	0	0	1	0
t_7	0	0	0	0	0	1	0	1
t_8	0	0	0	0	0	0	0	0

Data Object Dependency

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
t_1	0	1/4	1/4	1/4	1/4	0	0	1/2
t_2	1/4	0	3/4	3/4	5/4	1/2	1/2	3/2
t_3	1/4	3/4	0	3/2	5/4	1/4	1/4	1
t_4	1/4	3/4	3/2	0	5/4	1/4	1/4	1
t_5	1/4	5/4	5/4	5/4	0	1/2	1/2	3/2
t_6	0	1/2	1/4	1/4	1/2	0	5/4	3/2
t_7	0	1/2	1/4	1/4	1/2	5/4	0	3/2
t_8	1/2	3/2	1	1	3/2	3/2	3/2	0

Aggregation

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
t_1	0	5/4	1/4	1/4	1/4	0	0	1/2
t_2	1/4	0	7/4	3/4	5/4	1/2	1/2	3/2
t_3	1/4	3/4	0	5/2	5/4	1/4	1/4	1
t_4	1/4	3/4	3/2	0	9/4	5/4	1/4	1
t_5	1/4	5/4	9/4	5/4	0	1/2	1/2	3/2
t_6	0	1/2	1/4	1/4	1/2	0	9/4	3/2
t_7	0	1/2	1/4	1/4	1/2	9/4	0	5/2
t_8	1/2	3/2	1	1	3/2	3/2	3/2	0

Clustering



Microservice
candidates

Sources: *Object-aware Identification*, M.J. Amiri



Weakness

- Aggregation lacks mathematical evidence
- Weighting for read/write access does not consider inter-/intra service calls
 - Remote calls outweigh difference between read and write access
- Data flow only implicitly

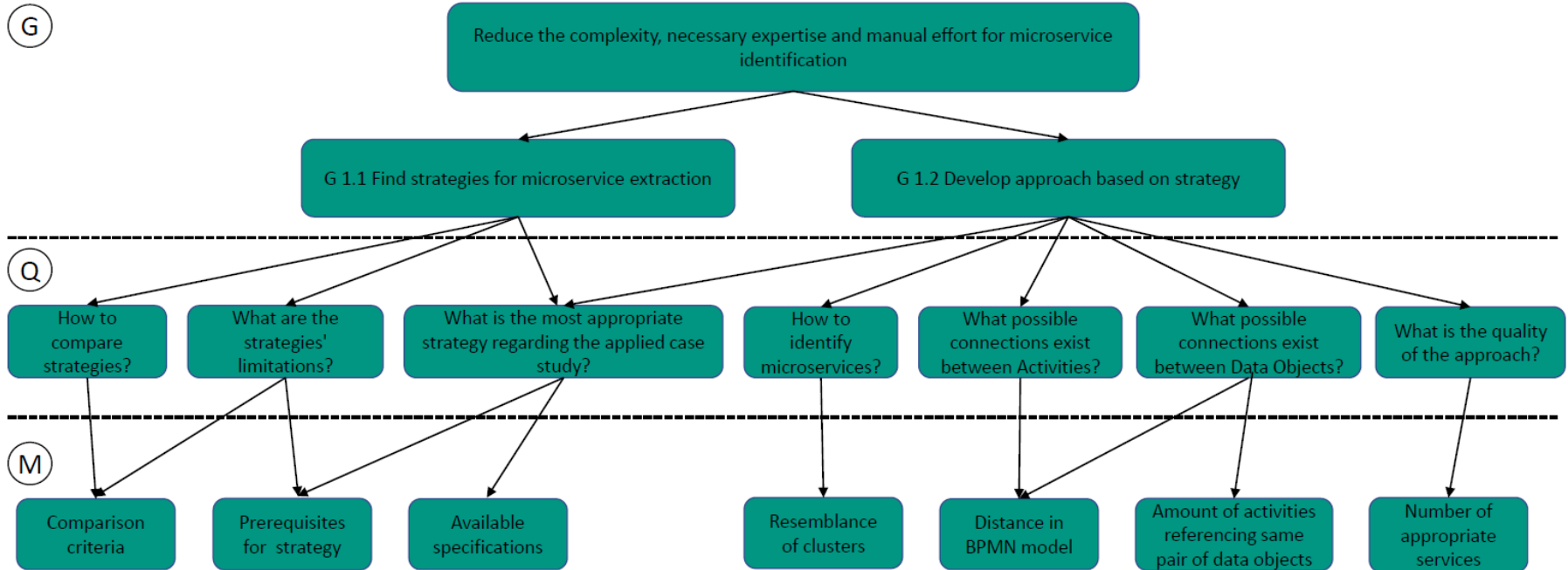


Blurred data- and control flow



Contributions

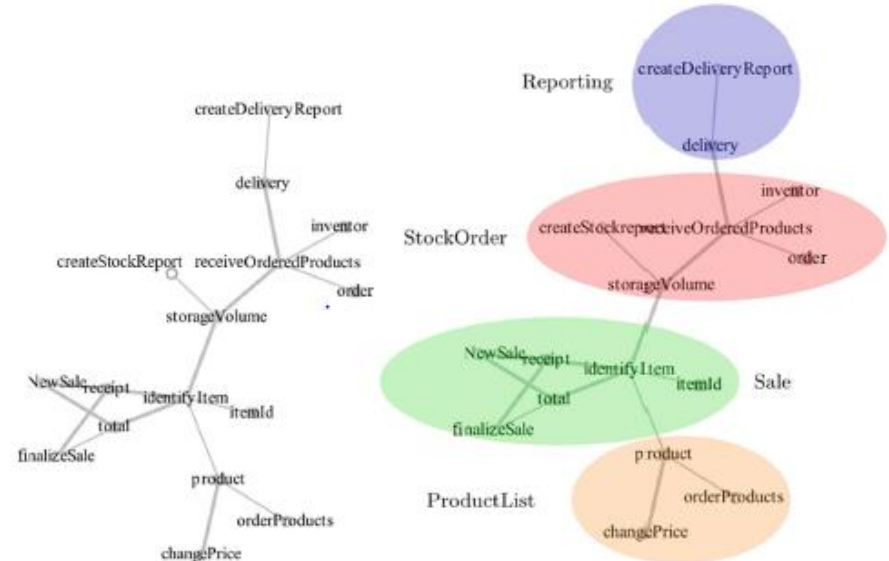
- Separate data object dependency and structural dependency
 - Cluster based on structural dependency as proposed
 - Cluster for object dependency based on new approach
 - ➡ Circumvent ambiguity of aggregation process
- New approach for data object dependency:
 - Distance Measure
 - Amount activities that access pair of objects
 - ...
- Match both set of clusters
 - Based on BPMN Model information
 - Merge & split possible



Source: Identifying Microservices Using Functional Decomposition, S. Tyszberowicz

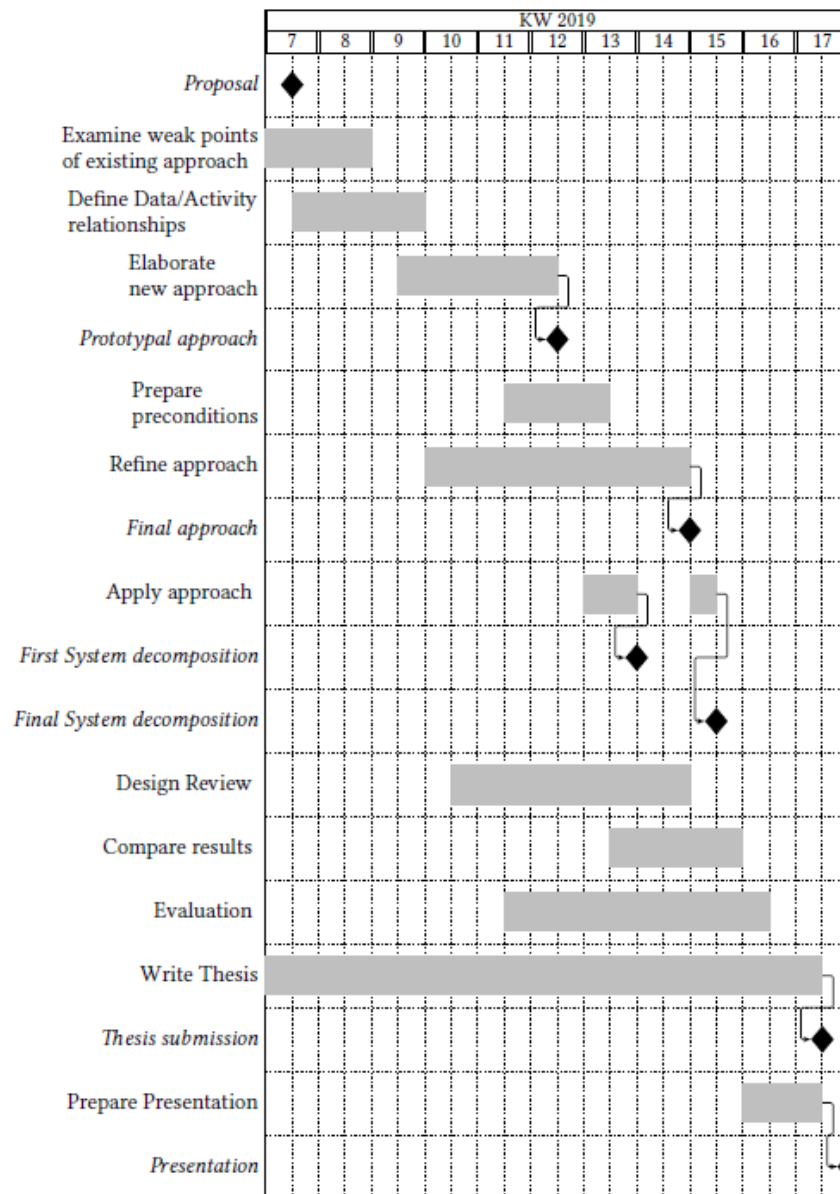
Evaluation: Results

- Apply to case study *CoCoME*
- Evaluate results
 - Amount of similar microservices
 - Assess the quality of the other identified services
- Evaluate limitations
- Provide outlook



Source: Identifying Microservices Using Functional Decomposition, S. Tyszberowicz

Summary



Motivation



Foundation



Related Work



Approach



Evaluation



Summary

WHEN YOUR BOSS TELLS YOU



Source: <https://twitter.com/shanselman/status/785985307816493057>