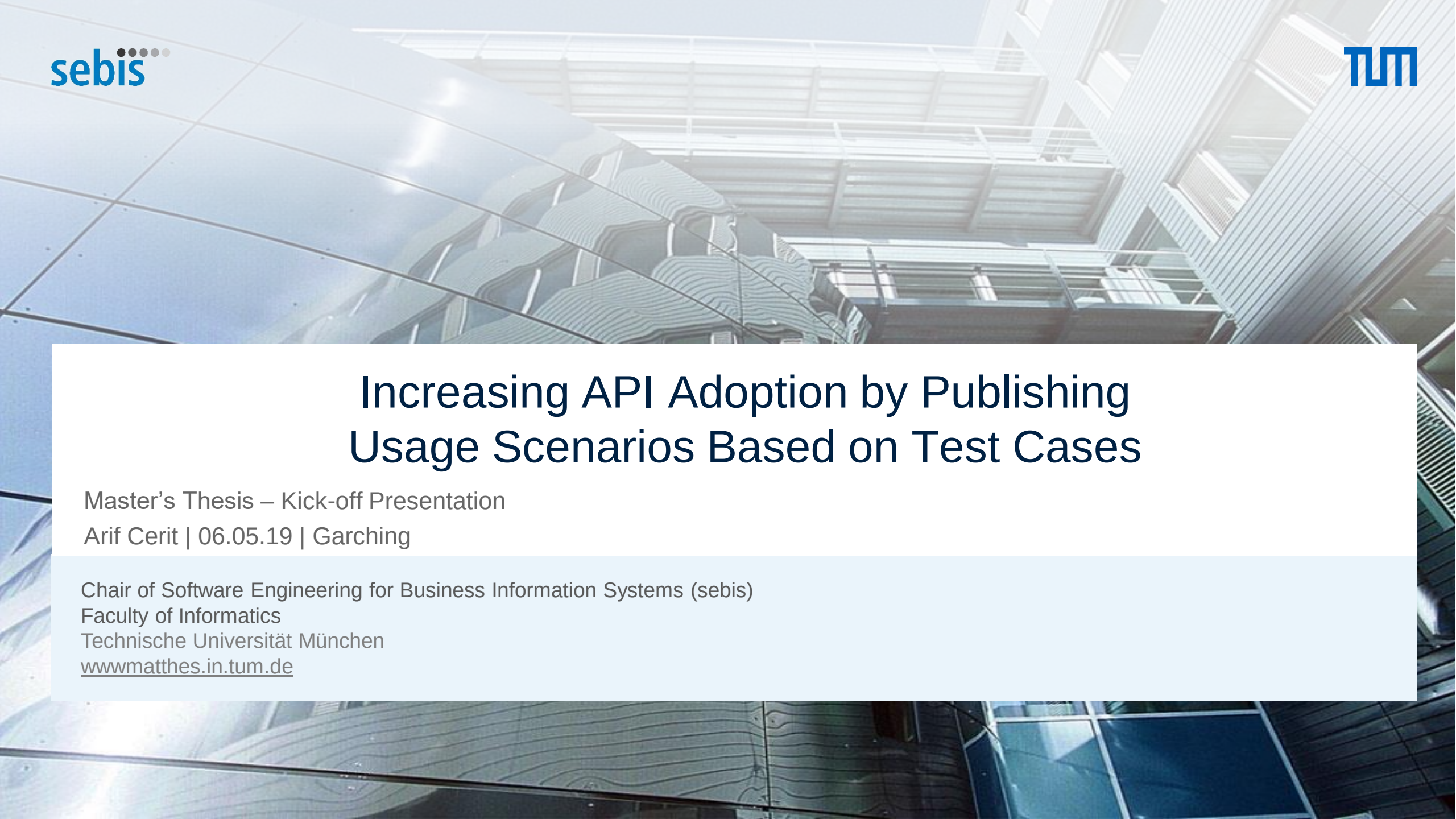


Increasing API Adoption by Publishing Usage Scenarios Based on Test Cases

Master's Thesis – Kick-off Presentation

Arif Cerit | 06.05.19 | Garching

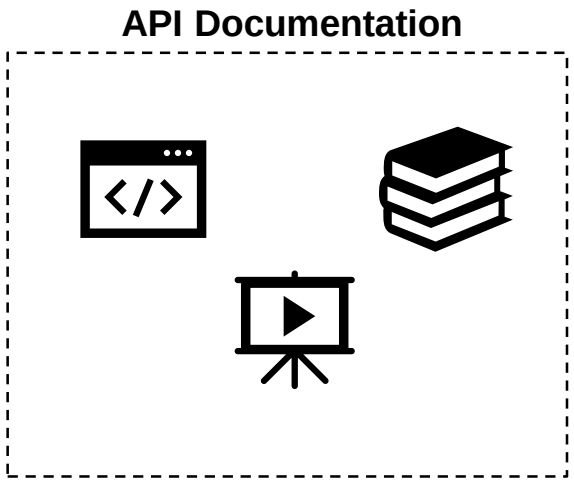
Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
wwwmatthes.in.tum.de

Outline

- Motivation
- Research Questions
- Approach
- Status Quo
- Next Steps

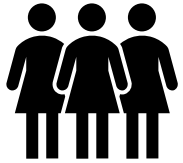
Motivation

Using and learning API‘s is hard



API Provider

API Spec



API Consumer

→ Information Asymmetry

Motivation

Using and learning API‘s is hard

Challenges for API Adoption

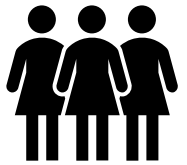
API Provider

No „look under the hood“ possible

Complex scenarios and journeys not covered [1]

Mostly syntactical/technical information [1]

Lack of high-quality content and examples [2] [3]



API Consumer

[1] Robillard & DeLine, 2011 | [2] Sohan et al., 2017 | [3] Hoffman & Strooper, 2003

Motivation

Creating an order

POST /commerce/api/users/max/orders?cartid=123

Preconditions

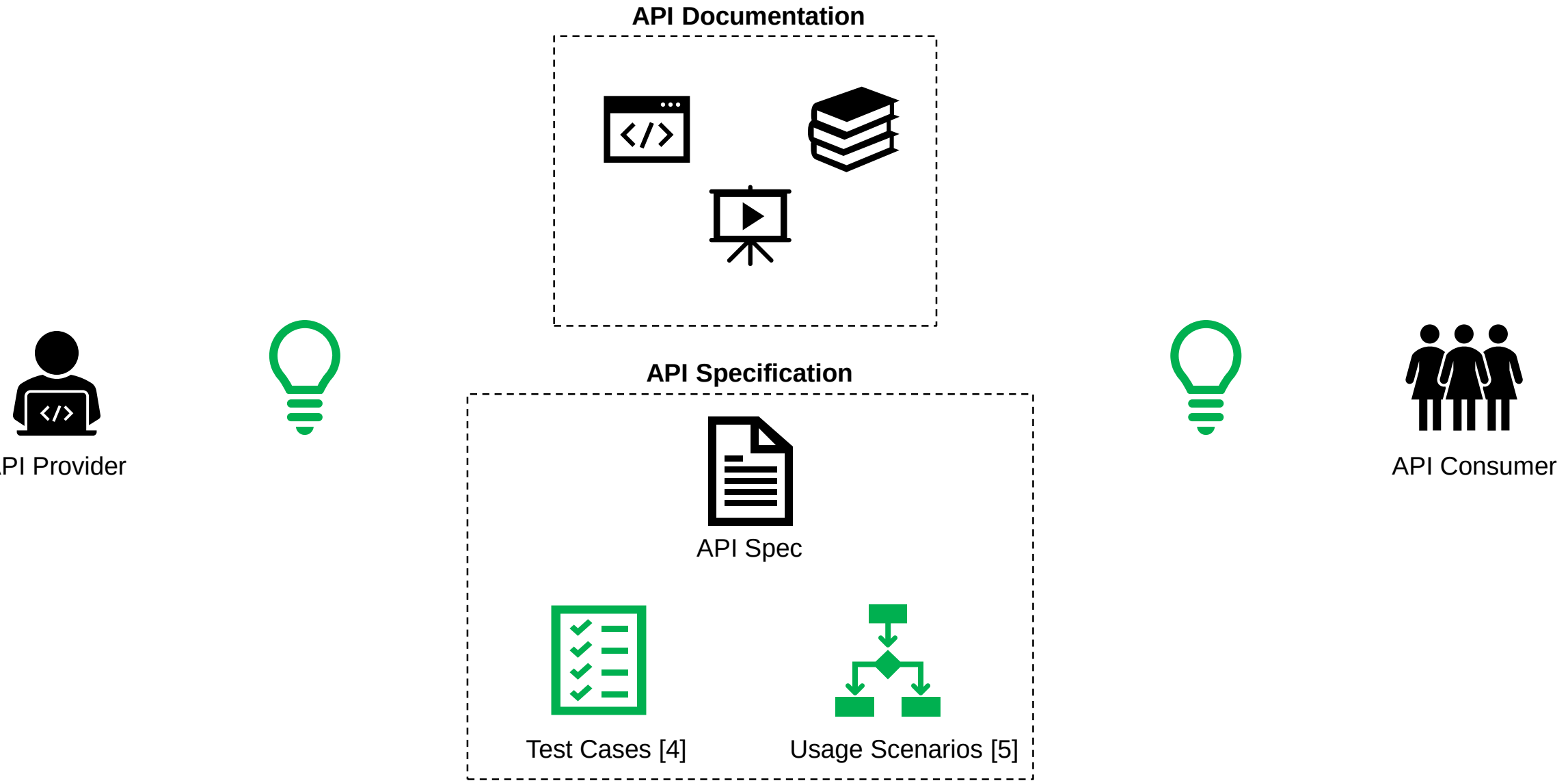
GET /commerce/api/users/max
GET /commerce/api/users/max/paymentdetails
GET /commerce/api/carts/123
…

Postconditions

POST /commerce/api/payments/validate
DELETE /commerce/api/carts/123
POST /commerce/api/delivery
…

Motivation

The missing link



[4] Nasehi & Maurer, 2010 | [5] Glassman et al., 2018

Research Questions

RQ 1 What are the **approaches** and **concepts** to expose and create API usage scenarios and test cases?

Literature Review + Interviews

RQ 2 What **semantic information** must be included in API usage scenarios and test cases?

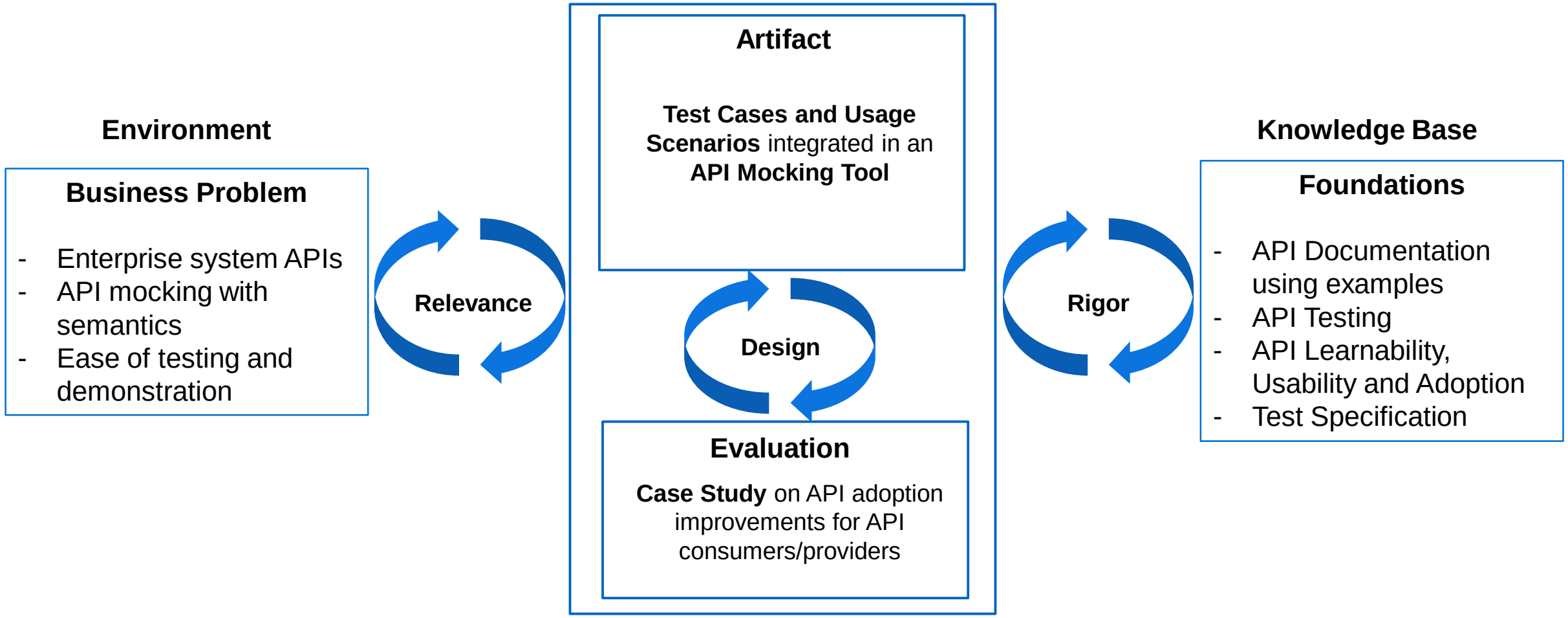
Interviews

RQ 3 How can **tool-supported publication** of API usage scenarios and test cases help increase **adoption** and **usability**?

Implementation + Case Study

Approach

Design Science Research [6] [7]



[6] Hevner et al., 2004 | [7] Peffers et al., 2007

Approach
Environment

- **SAP Customer Experience – Cloud Business Group**

- **Domain:** E-Commerce, CRM, Marketing

- **Project**
  - Orchestration of APIs & Events
  - Extensibility and integration
  - Unified management plane (cockpit)

Approach
Environment

C/4 CORE
~ 260 employees

PO

Dev

Project A - Cockpit
(3 teams)
Walldorf

Project B – Extension/Orchestration
(11 teams)
Munich, Gliwice (Poland)

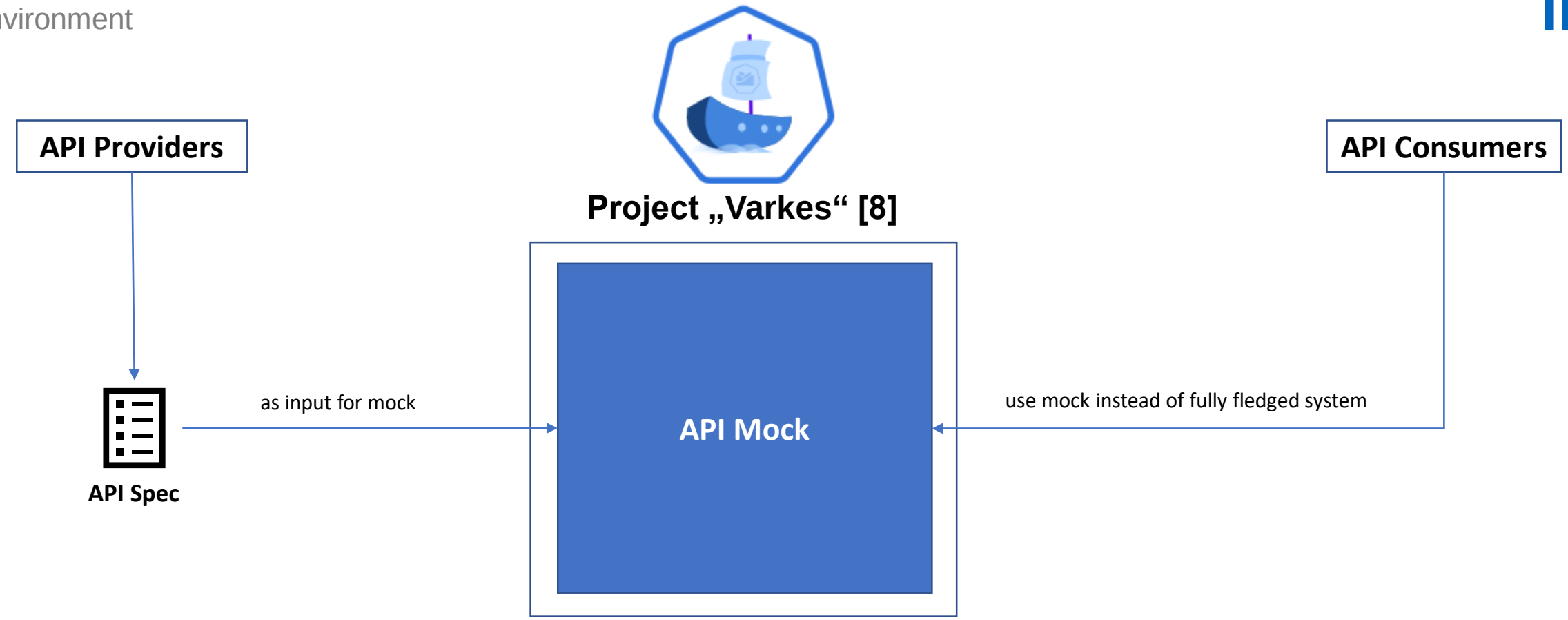
PO

Dev

SW Architect

EA

Approach
Environment



\+ covers all endpoints
\+ responds with examples from spec

\- bad spec → bad mock
\- misleading behavior

[8] https://github.com/kyma-incubator/varkes

Approach
Knowledge Base

API Documentation

(Code) Examples [3] [9] [10]

Specification-based [11]

Usage scenarios [2]

API Usability & Adoption

Learning challenges [1] [12]

Evaluation of usability [13]

API Mocking

Market Research
- Import functionality
- Customizability
- Event-driven …

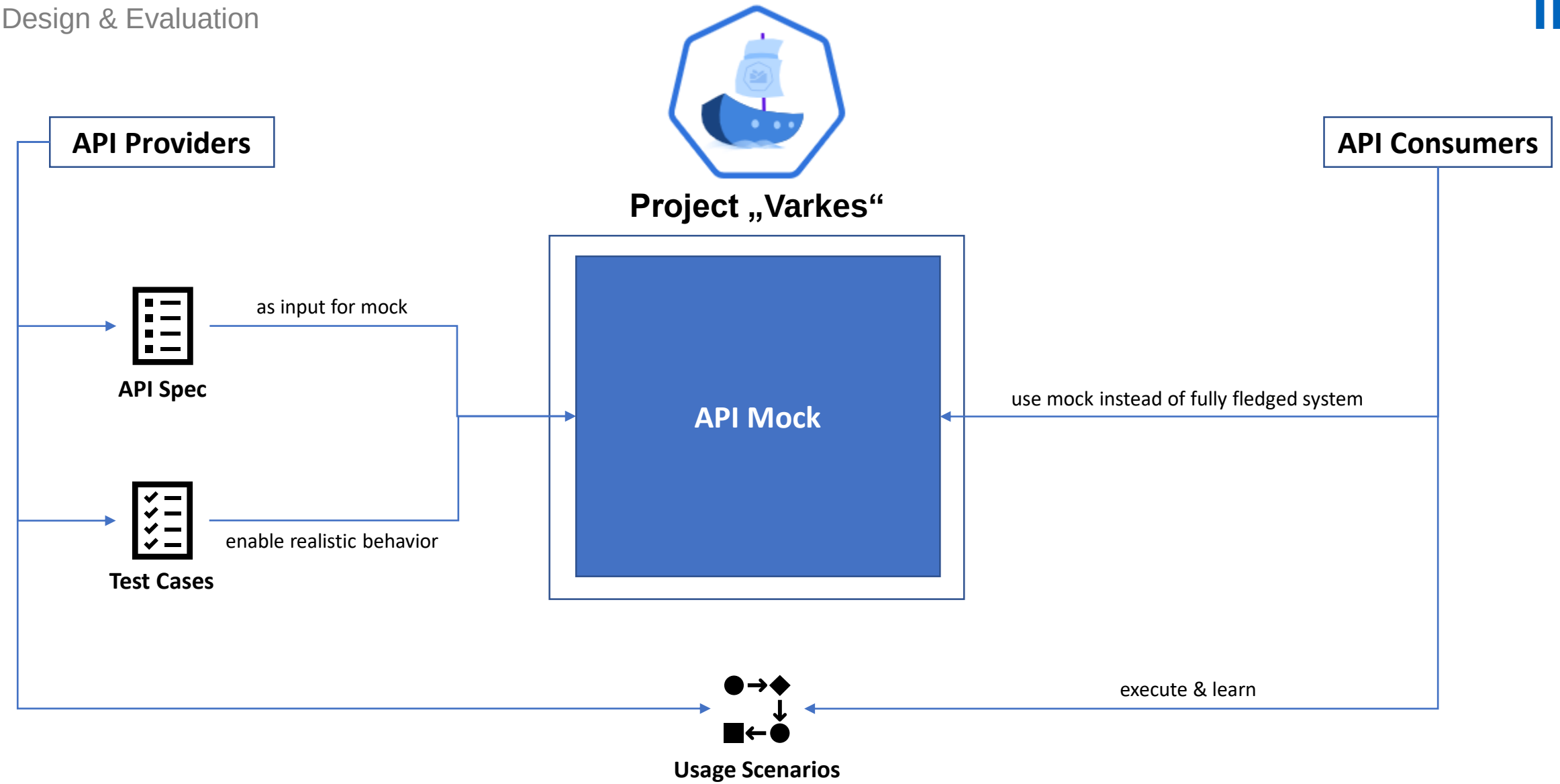
Related fields
- Service Virtualization
- API Stubs

→ Mostly referring to software library APIs!

[9] Hoffman & Strooper, 2000 | [10] Sohan et al., 2017 | [11] Ed-douibi et al., 2018 | [12] Robillard, 2009 | [13] Farooq et al., 2010

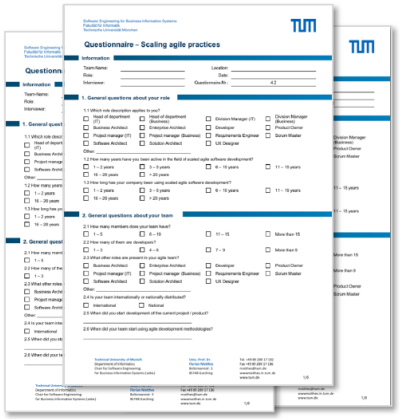
Approach

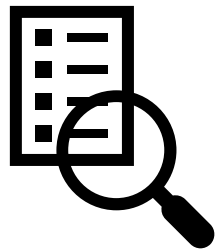
Design & Evaluation



Evaluation Case Study [14]

Design → Preparation → Data Collection → Data Analysis





[14] Yin, 2014

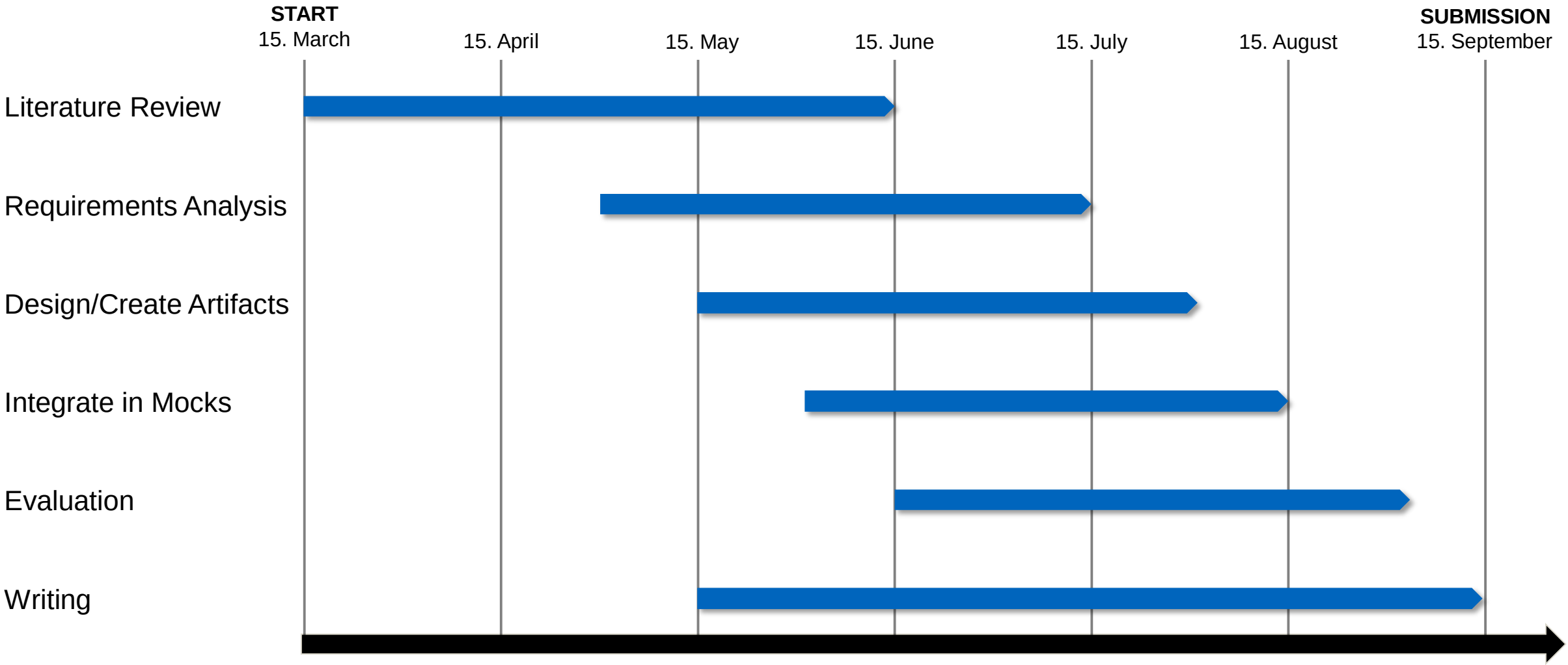
Status Quo

- **Identified concepts in API Documentation and API Testing**
  - Examples, Test Cases and Scenarios
  - Specification-based, Mocking, E2E Testing

- **Requirements for approach**
  - Unstructured interviews with two architects
  - „Most/All of the information should be conveyed using one specification.“
  - „We need good mocks for our demos and experiments with enterprise systems.“

- **Analyzed API Mocking tools**

Next Steps


START
15. March
15. April
15. May
15. June
15. July
15. August
SUBMISSION
15. September
Literature Review
Requirements Analysis
Design/Create Artifacts
Integrate in Mocks
Evaluation
Writing

Literature

[1] Robillard, Martin P.; DeLine, Robert (2011): A field study of API learning obstacles. In: Empirical Software Engineering 16 (6), S. 703–732.

[2] Sohan, S. M.; Maurer, Frank; Anslow, Craig; Robillard, Martin P. (2017): A study of the effectiveness of usage examples in REST API documentation. In: IEEE Symposium on Visual Languages and Human-Centric Computing, S. 53–61.

[3] Hoffman, Daniel; Strooper, Paul (2003): API documentation with executable examples. In: Journal of Systems and Software 66 (2), S. 143–156.

[4] Nasehi, Seyed Mehdi; Maurer, Frank (2010): Unit tests as API usage examples. In: IEEE International Conference on Software Maintenance (ICSM), S. 1–10.

[5] Glassman, Elena L.; Zhang, Tianyi; Hartmann, Björn; Kim, Miryung (2018): Visualizing API Usage Examples at Scale. In: CHI Conference on Human Factors in Computing Systems, S. 1–12.

[6] Hevner, Alan; March, Salvatore; T, Salvatore; Park; Park, Jinsoo; Ram; Sudha (2004): Design Science in Information Systems Research. In: Management Information Systems Quarterly 28, 75.

[7] Peffers, Ken; Tuunanen, Tuure; Rothenberger, Marcus A.; Chatterjee, Samir (2007): A Design Science Research Methodology for Information Systems Research. In: Journal of Management Information Systems 24 (3), S. 45–77.

[8] Project “Varkes”, https://github.com/kyma-incubator/varkes.

[9] Hoffman, Daniel; Strooper, Paul (2000): Prose+test cases=specifications. In: International Conference TOOLS. IEEE Computer Society, S. 239–250.

[10] Sohan, S. M.; Anslow, Craig; Maurer, Frank (2017): Automated example oriented REST API documentation at Cisco. In: 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE).

[11] Ed-douibi, Hamza; Canovas Izquierdo, Javier Luis; Cabot, Jordi (2018): Automatic Generation of Test Cases for REST APIs: A Specification-Based Approach. In: IEEE EDOC, S. 181–190.

[12] Robillard, Martin P. (2009): What Makes APIs Hard to Learn? Answers from Developers. In: IEEE Softw. 26 (6), S. 27–34.

[13] Farooq, Umer; Welicki, Leon; Zirkler, Dieter (2010): API usability peer reviews: a method for evaluating the usability of application programming interfaces: ACM.

[14] Yin, Robert K. (2014): Case study research. Design and methods. 5. edition. Los Angeles, London, New Delhi, Singapore, Washington, DC: SAGE.

B. Sc.

Arif Cerit

arif.cerit@tum.de

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for Business Information Systems

Boltzmannstraße 3
85748 Garching bei München

wwwmatthes.in.tum.de

Backup

Market Research

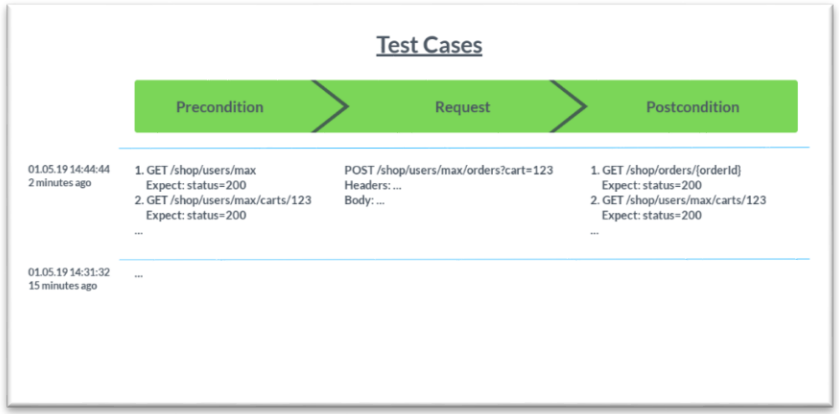
Design & Evaluation

Tools on Market	Requirement	Varkes
Stoplight, Restlet, Postman, SoapUI	**Specification as Input**	X
Stoplight, Restlet, SoapUI, MockServer	**Customizable Response**	X
?	**Events (AsyncAPI)**	X
Stoplight, Restlet, Postman, SoapUI, WireMock	**Usage Scenarios**	-
?	**Support for OpenAPI and OData**	X
? (most likely none)	**Connection to SAP XF**	X

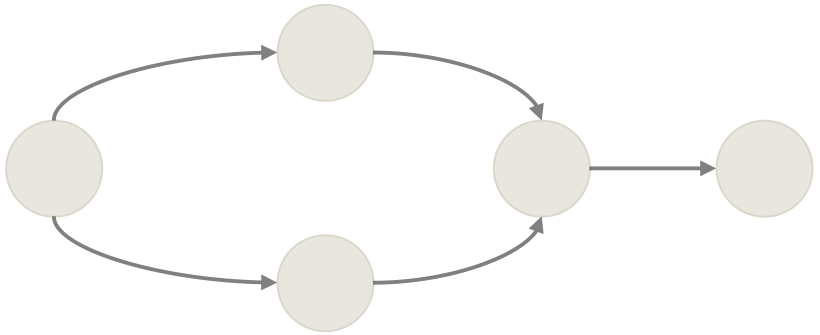
Design

Artifact 1: Test Cases

Preconditions → **API Request** → Postconditions



Artifact 2: Usage Scenarios



Approach
Knowledge Base

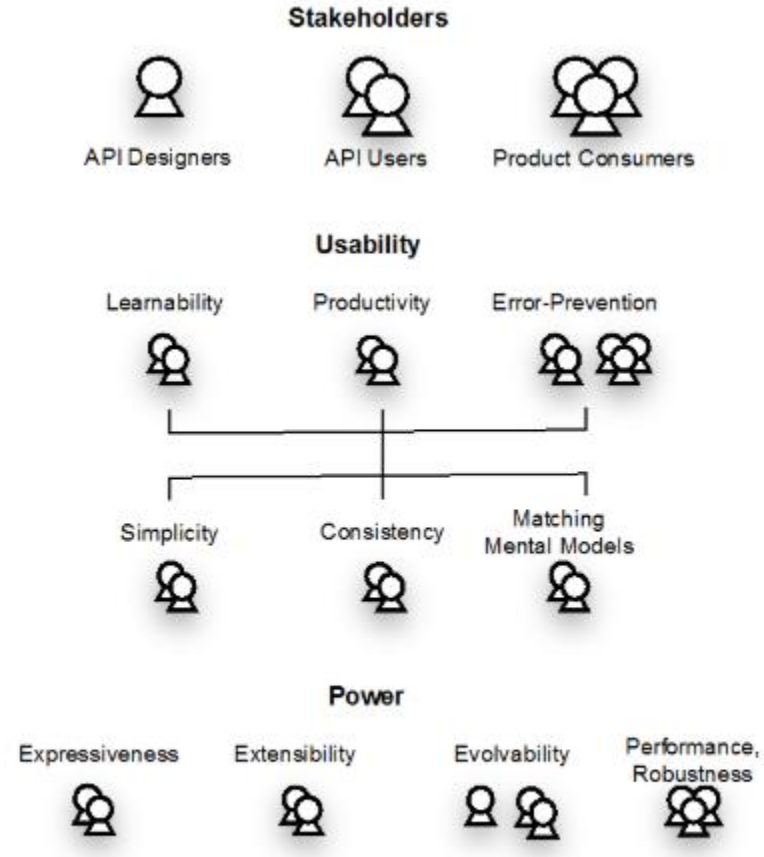


Figure 1. Quality attributes of APIs, and the stakeholders most affected by each quality.

API Learning Obstacles

Table 1

Response categories for API learning obstacles

Main category	Subcategories/descriptions		Associated respondents
Resources	Obstacles caused by inadequate or absent resources for learning the API (for example, documentation)		50
	Examples	Insufficient or inadequate examples	20
	General	Unspecified issues with the documentation	14
	Content	A specific piece of content is missing or inadequately presented in the documentation (for example, information about all exceptions raised)	12
	Task	No reference on how to use the API to accomplish a specific task	9
	Format	Resources aren't available in the desired format	8
	Design	Insufficient or inadequate documentation on the high-level aspects of the API such as design or rationale	8
Structure	Obstacles related to the structure or design of the API		36
	Design	Issues with the API's structural design	20
	Testing and debugging	Issues related to the API's testing, debugging, and runtime behavior	10
Background	Obstacles caused by the respondent's background and prior experience		17
Technical environment	Obstacles caused by the technical environment in which the API is used (for example, heterogeneous system, hardware)		15
Process	Obstacles related to process issues (for example, time, interruptions)		13