Identification and Evaluation of Incentive Mechanisms for Opening Internal Systems via Partner APIs

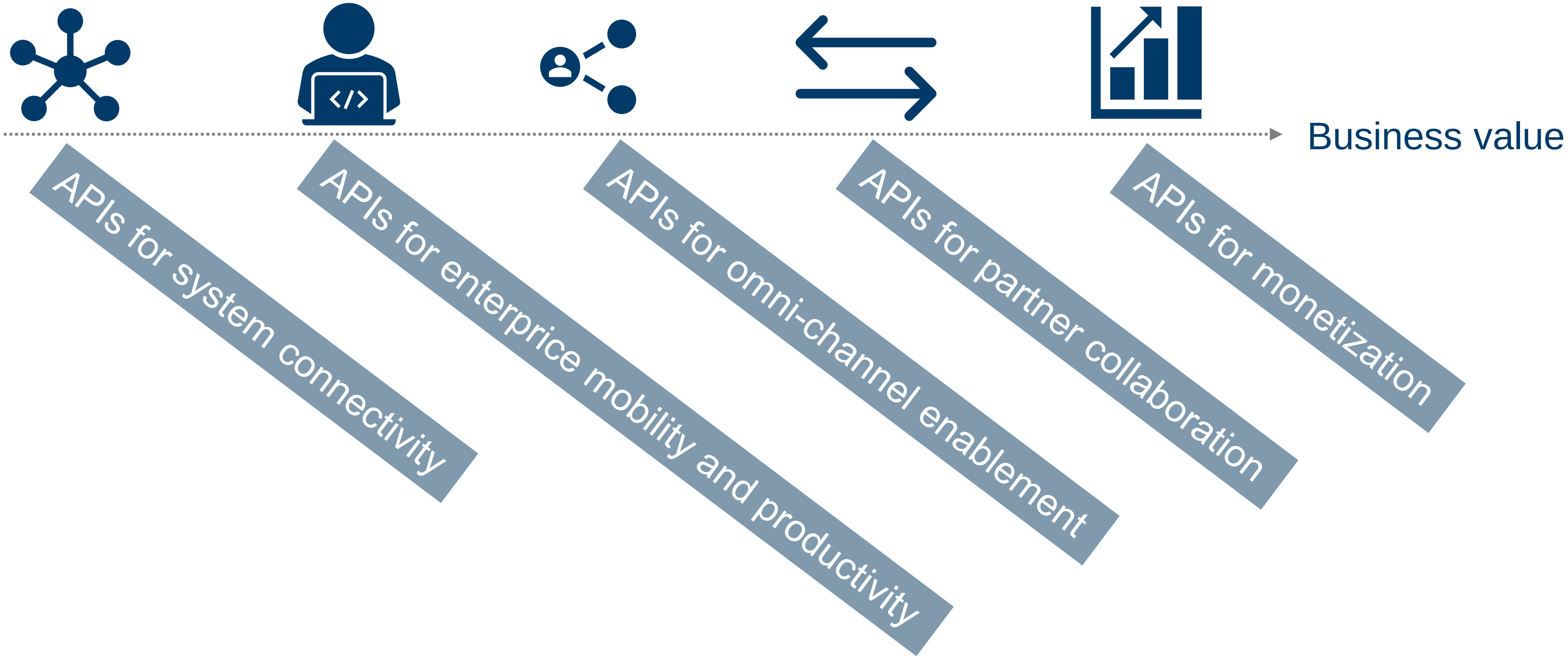
Irena Stoilova, 06.07.2020, München

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

Outline

- Motivation
- Problem statement
- Research Questions
- Approach
- Results
- Summary and Outlook

Motivation – the API value chain



Business value

APIs for system connectivity

APIs for enterprice mobility and productivity

APIs for omni-channel enablement

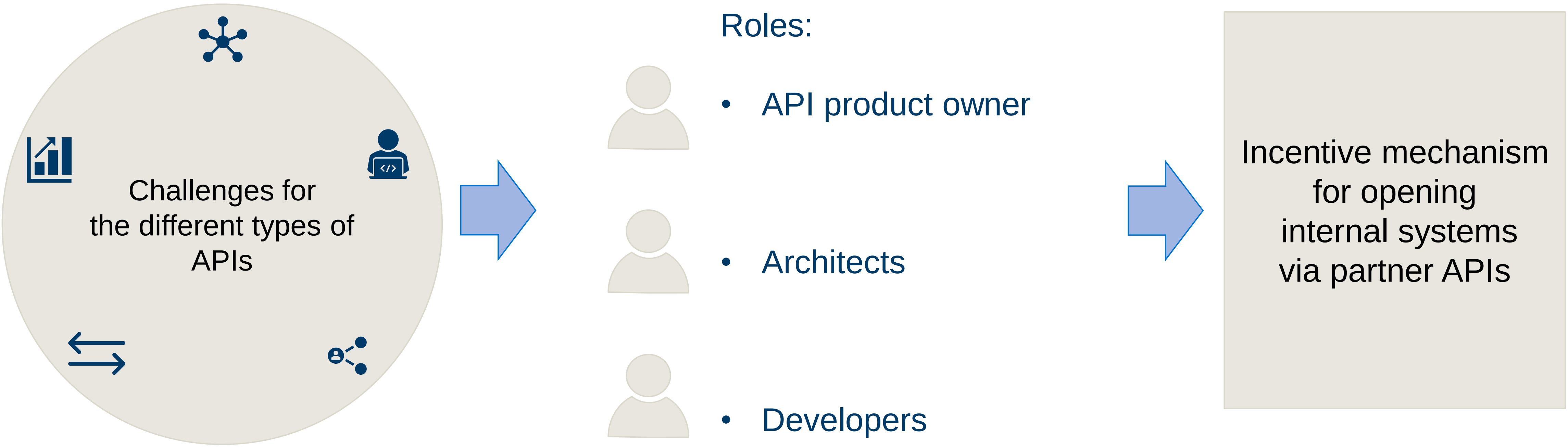
APIs for partner collaboration

APIs for monetization

Source: Weir, Luis. *Enterprise API Management: Design and deliver valuable business APIs*. Packt Publishing Ltd, 2019.

Problem Statement

From a development team to an API product team:



Challenges for the different types of APIs

Roles:

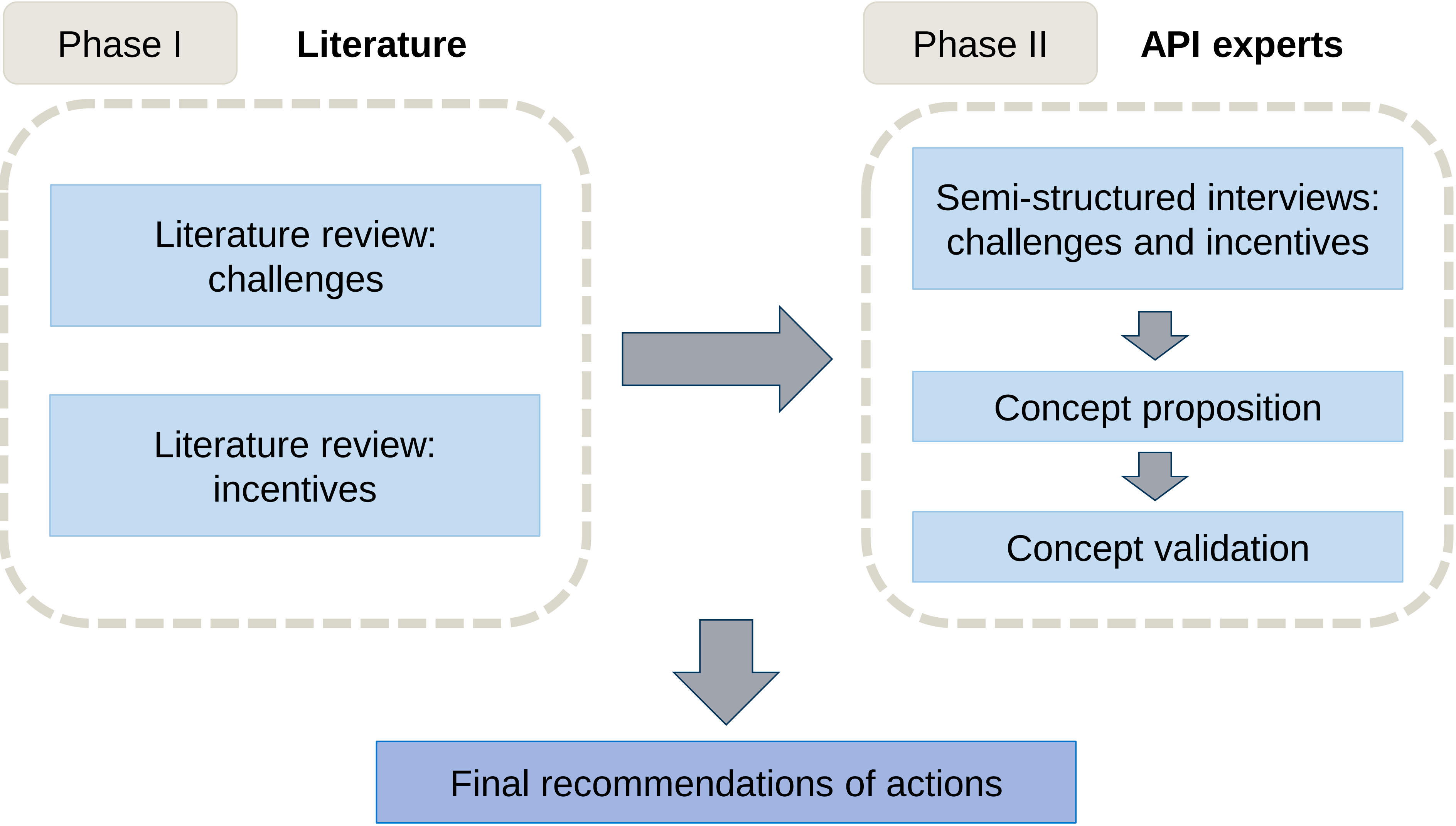
- API product owner
- Architects
- Developers

Incentive mechanism for opening internal systems via partner APIs

Research Questions

1. What are the challenges for providing partner APIs in internal solutions?
2. What are existing incentive mechanisms in literature motivating teams to provide partner APIs?
3. What are incentives for providing APIs or API platforms in a large IT organization?
4. What are recommendations for incentivizing developers and architects to provide partner APIs?

Research Approach



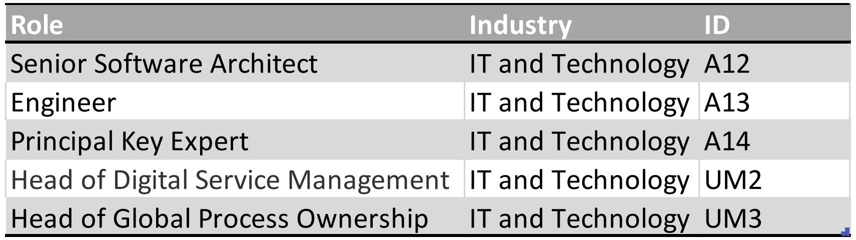
Phase I
Literature
Literature review: challenges
Literature review: incentives
Phase II
API experts
Semi-structured interviews: challenges and incentives
Concept proposition
Concept validation
Final recommendations of actions

Interviews – Design

Interview phase:

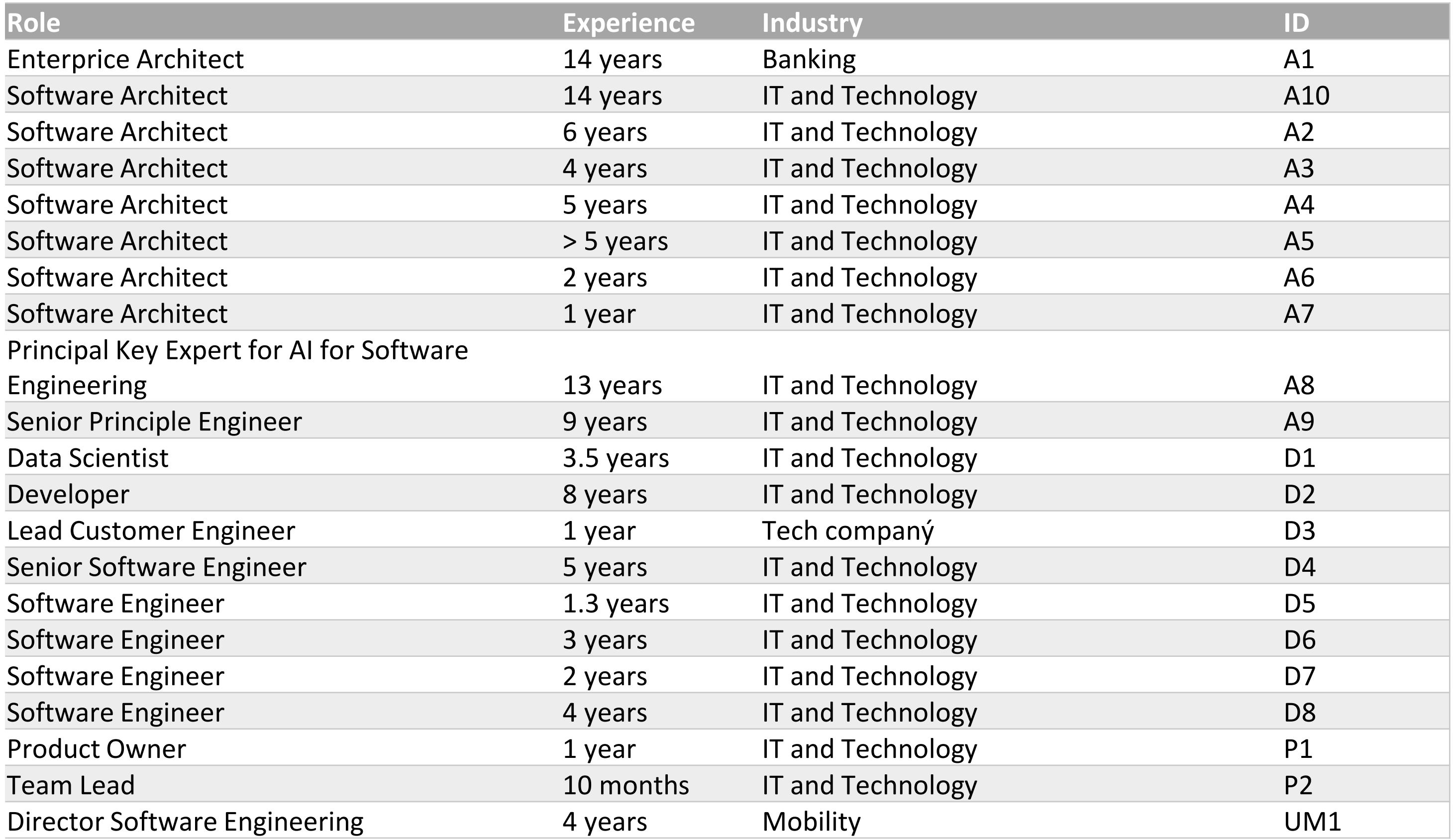
- Exploratory interviews with 5 experts
- Semi-structured interviews with 21 professionals

Table 1. Participants in the exploratory interviews



Role	Industry	ID
Senior Software Architect	IT and Technology	A12
Engineer	IT and Technology	A13
Principal Key Expert	IT and Technology	A14
Head of Digital Service Management	IT and Technology	UM2
Head of Global Process Ownership	IT and Technology	UM3

Table 2. Participants in the semi-structured interviews



Role	Experience	Industry	ID
Enterprise Architect	14 years	Banking	A1
Software Architect	14 years	IT and Technology	A10
Software Architect	6 years	IT and Technology	A2
Software Architect	4 years	IT and Technology	A3
Software Architect	5 years	IT and Technology	A4
Software Architect	> 5 years	IT and Technology	A5
Software Architect	2 years	IT and Technology	A6
Software Architect	1 year	IT and Technology	A7
Principal Key Expert for AI for Software Engineering	13 years	IT and Technology	A8
Senior Principle Engineer	9 years	IT and Technology	A9
Data Scientist	3.5 years	IT and Technology	D1
Developer	8 years	IT and Technology	D2
Lead Customer Engineer	1 year	Tech companý	D3
Senior Software Engineer	5 years	IT and Technology	D4
Software Engineer	1.3 years	IT and Technology	D5
Software Engineer	3 years	IT and Technology	D6
Software Engineer	2 years	IT and Technology	D7
Software Engineer	4 years	IT and Technology	D8
Product Owner	1 year	IT and Technology	P1
Team Lead	10 months	IT and Technology	P2
Director Software Engineering	4 years	Mobility	UM1

Challenges in API development and maintenance

Similarities between results of literature review and interview results

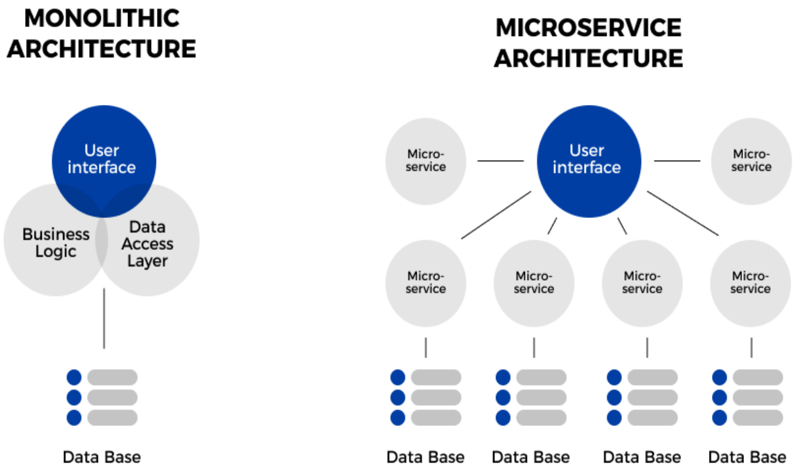
Developers education	API design and implementation on-hands experience
Usability	provide interface with high level of usability
Documentation	automatically create high quality documentation out of implementation
Management	important for high customer satisfaction
API platform	establishment is important to ensure visibility
Monitoring	lack of tools; correct analysis of monitoring data

Challenges in API development and maintenance (continued)

Diffenreces between results of literature review and interview results

Topic	Literature review	Industry practice
API platform	continuous value delivery	too small developer community
API strategy	intentionally provide assests via APIs	question the need of an API
Documentation	discoverable	understandable
Deployment	lack of dedicated platform	access obstacles (e.g. for cloud platforms)
Security	finding the right technology	organisational challenges
Deprecation	strategy to be estalished	project specific

Projects and Industry Partners



1. Multinational IT organization (main industry partner):
   - Migrating from monolith to microservice-based architecture
   - API Gateway
   - API Developer Portal
2. Multinational technology company
3. Finance
4. Mobility

Source: https://divante.com/blog/monolithic-architecture-vs-microservices/

Project descriptions

API Platform

- Platform centered to one department
- APIs related to IoT topics
- Existing incentives for external developers

API Gateway

- Designed to ease app developers
- Collects engineering and simulation, operations and monitoring and maintenance data

API Developer Portal

- Platform aimed to all divisions
- Paused due to lack of funding
- Goal: bring transparency

Existing incentive mechanisms – literature review

Industry status	Project-bound establishment	Need identified	✕ Missing
Project Management	Creating awareness [1]	Visible success; meeting targets; bottom-up initiatives and top-down commitment [1], [2], [3] Innovation; Shift mindset towards IT value [4]	Marketing strategy; [1]
Recognition	Recognition by senior management [2]	Receiving gratitude - from peers or management; ownership culture; [2], [5]	Challenges with prizes, badges, gifts [1]
Technical		Resources; [2] Guidelines; [6] Easy software integration into platform; sample code; quality tech support; [1]	
Self-development		Community and collaboration; Exchanging knowledge; Creating new forms of cooperation [1], [5]	Developer programs available on premise; Free downloads for self-development; [1]

Existing incentives in large IT organisations?

Granularity

- Complexity reduction of legacy systems
- Available support from open source community, documentation
- Easy maintenance of systems: deploying new features, tracing errors and calls
- Transparency of usage: easier optimization of the system

APIs as a direction for new features

- Easily enter an ecosystem
- Fast realization of demands from various businesses
- Actively engage customers with new features

Transparency

- Better understanding of the business process
- Direct evidence of work pay off through API usage data
- New future directions for teams‘ development

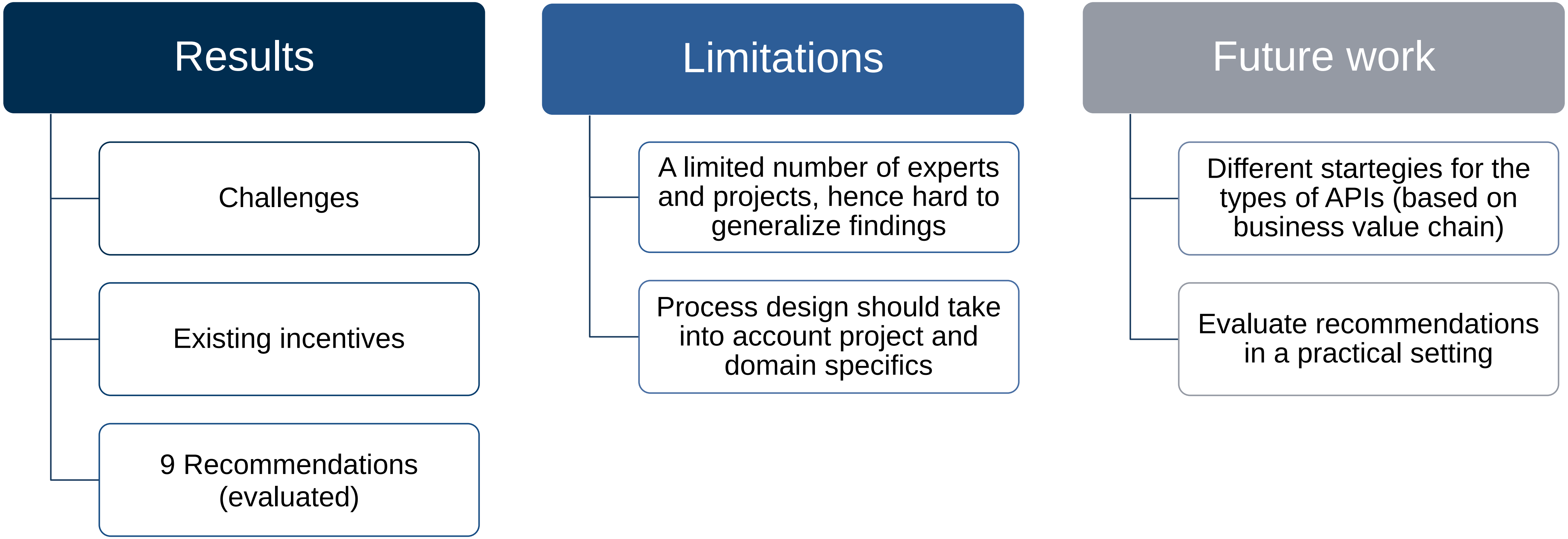
Recommendations of actions

1. **Platform for data sharing** - data description, data owner, access to data
2. **Communication** - plan efficient communication between stakeholders and developers to get feedback on time. Work together with the business/person with understanding of the domain on the design
3. **Top management initiatives** - create awareness within the organization
4. **Project organization** - organize the people in a project-based team; think domain-driven; think as a player within an ecosystem; encourage ownership culture.
5. **Managed API Gateway** - key point is to enable easy authentication; others are e.g. availability, security, version control

Recommendations of actions (continued)

6. **Guidelines** - create best practices and standards to be followed within the organization; keep re-usability in mind in the design phase
7. **Development tools and environment** - enable developers to publish quickly, securely and in a managed way; provide tools for API development/maintenance
8. **Monetization strategy** - readily designed by the business unit
9. **Speed** - key factor for both developers and customers (enable developers to create APIs fast; self-service for customers with little technical background)

Limitations & Outlook



Results

- Challenges
- Existing incentives
- 9 Recommendations (evaluated)

Limitations

- A limited number of experts and projects, hence hard to generalize findings
- Process design should take into account project and domain specifics

Future work

- Different startegies for the types of APIs (based on business value chain)
- Evaluate recommendations in a practical setting

Irena Stoilova

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

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289. 17132
Fax +49.89.289.17136

Irena.stoilova@tum.de
wwwmatthes.in.tum.de

Bibliography

[1] E. Schindler, “Building a developer ecosystem: What vendors do to attract you to their platforms,” 2013. [Online]. Available: https://www.itworld.com/article/2704738/building-a-developer-ecosystem--what-vendors-do-to-attract-you-to-their-platforms.html. [Accessed: 17-Feb-2020].

[2] N. Baddoo and T. Hall, “Motivators of Software Process Improvement: An analysis of practitioners’ views,” *J. Syst. Softw.*, vol. 62, no. 2, pp. 85–96, 2002.

[3] D. R. Agrawal, “If Open Source Code Is a Public Good, Why Does Private Provision Work (Or Does It),” *LBJ J. Pub. Aff.*, vol. 18, p. 55, 2005.

[4] J. Lee, “The future of enterprise computing,” 2013, pp. 1–1.

[5] E. Haruvy, W. Fang, and S. Chakravarty, “Incentives for Developers’ Contributions and Product Performance Metrics in Open Source Development: An Empirical Exploration,” Apr. 2005.

[6] J. A. Halderman, “To strengthen security, change developers’ incentives,” *IEEE Secur. Priv.*, vol. 8, no. 2, pp. 79–82, Mar. 2010.

Concept evaluation

Data distribution, data latency, data security

Some existing best practices, often team-dependent

Positively evaluated; creating awareness about API platforms

Positively evaluated; experienced teams also need support for new designs

Positively evaluated; important for security

Standardization might not be applicable on a large scale; best practices exist in community

Tools for testing, mocking, module concept development needed

Initial awareness instead of whole strategy; rest is project-based

Positively evaluated; additionally: API user experience design