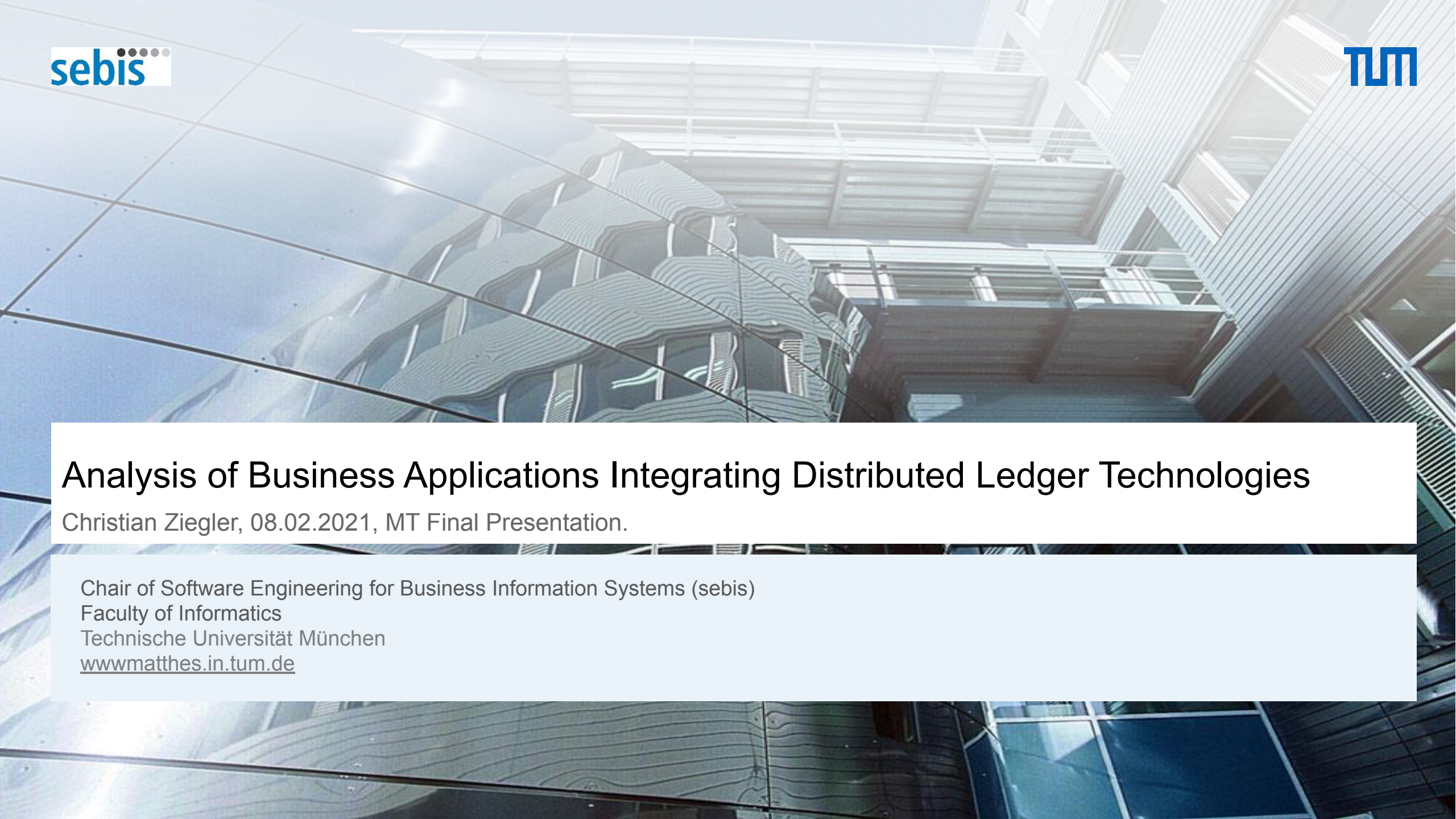


Analysis of Business Applications Integrating Distributed Ledger Technologies

Christian Ziegler, 08.02.2021, MT Final Presentation.

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

Outline

Motivation
- Evolution of DLT
- Academic Publications

Research Approach
- Overview
- Research Questions

Results
- Feature Catalog
  - Functional Features
  - Non-Functional Features
- Statistical Evaluation of the Use-Cases
  - Feature Usage
  - Platform Distribution
- Evaluation of the Reasonableness of DLT Implementations
- Future Work and Conclusion

Motivation – Evolution of DLT

Bitcoin

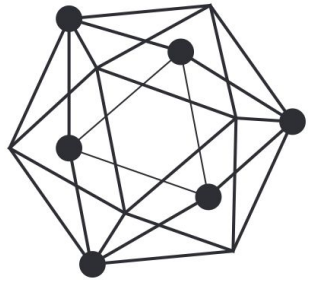
“Transfer money without a third party”

7187 Cryptocurrencies are active today [1]

Ethereum

“Execute code on a global virtual machine”

More than 2 million Smart Contracts deployed on Ethereum in November 2020 [2]



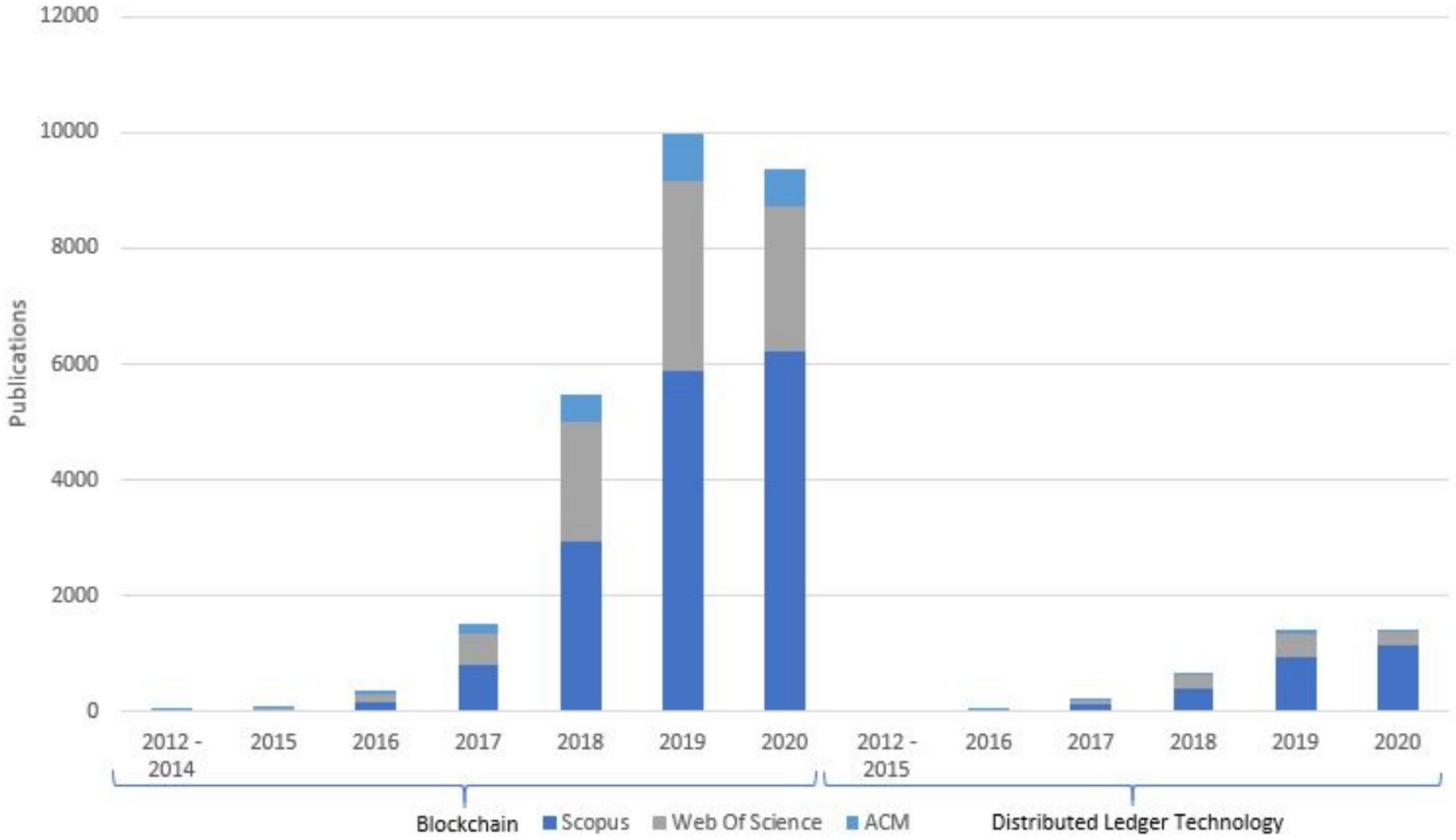
Hyperledger

“Build your own private blockchain for your enterprise system with all its benefits (and downsides...)”

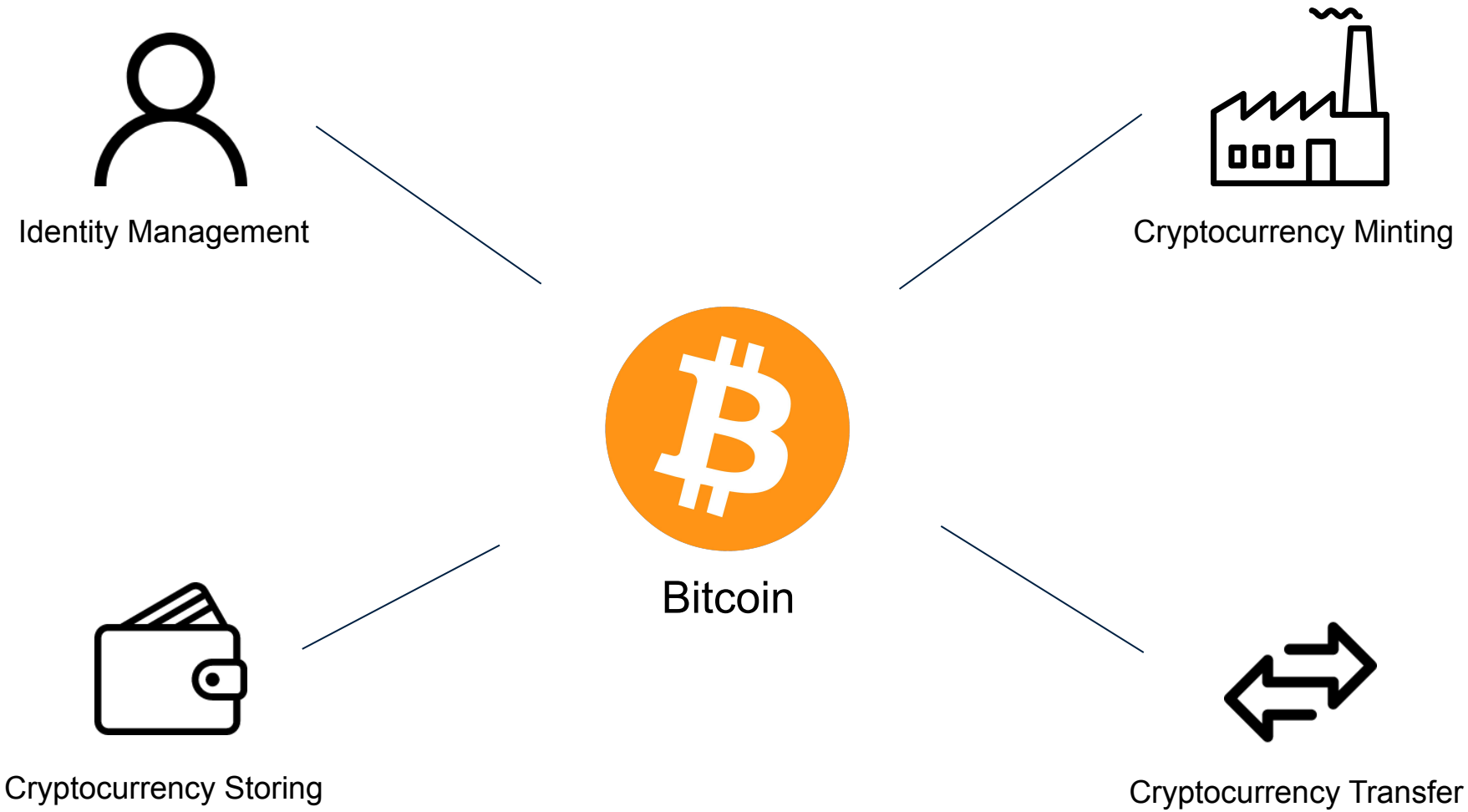
Hyperledger collaborates with more than 250 companies [3]

[1] https://coinmarketcap.com/all/views/all/
[2] https://explore.duneanalytics.com/embed/query/329/visualization/515?api_key=RqqCse8wDeW3MZKeZFOab2Ju8QA5Q8Itnr1cbLgk
[3] https://www.hyperledger.org/

Motivation – Academic Publications



Motivation – Evolution of DLT



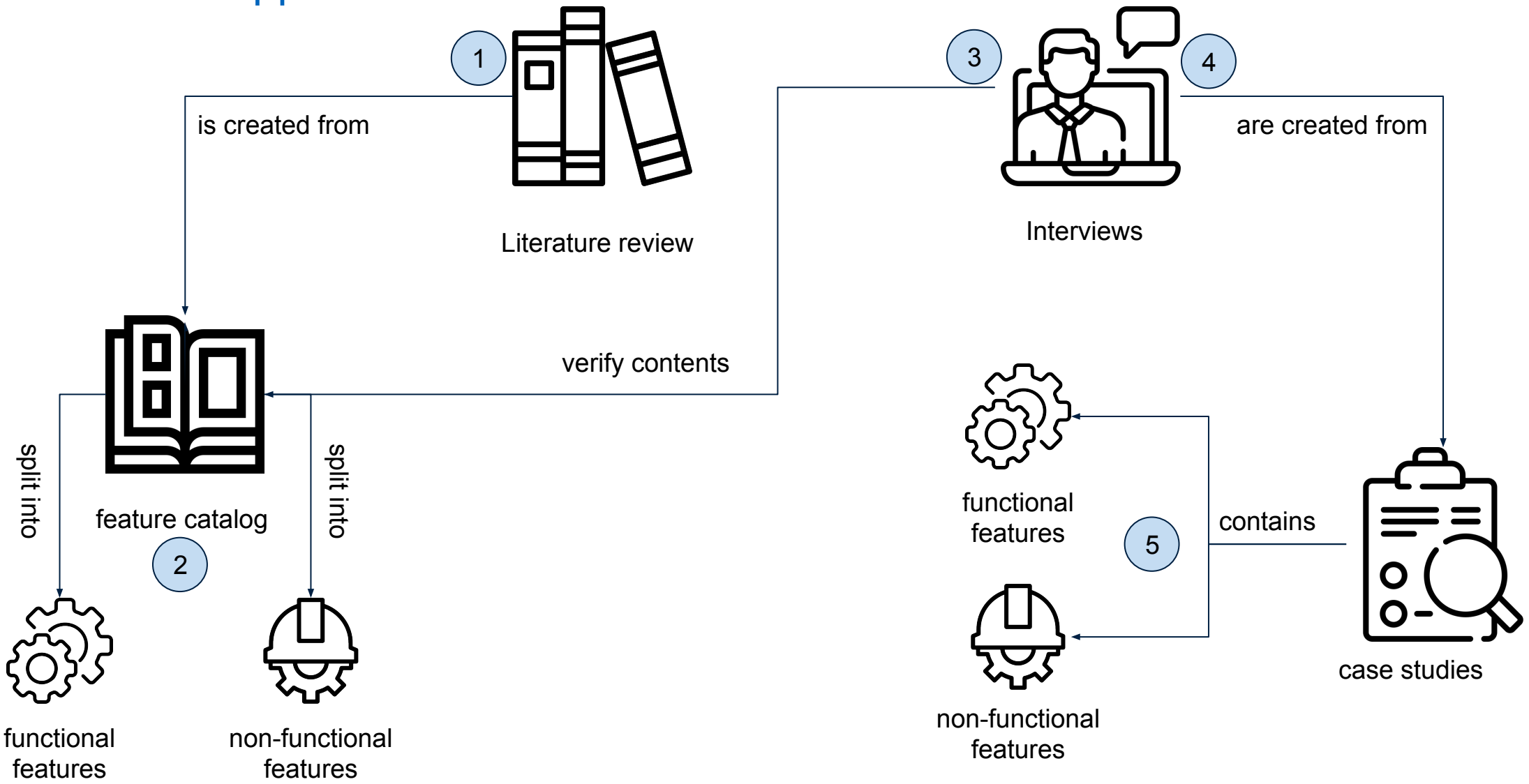
Outline

Motivation
- Evolution of DLT
- Academic Publications

Research Approach
- Overview
- Research Questions

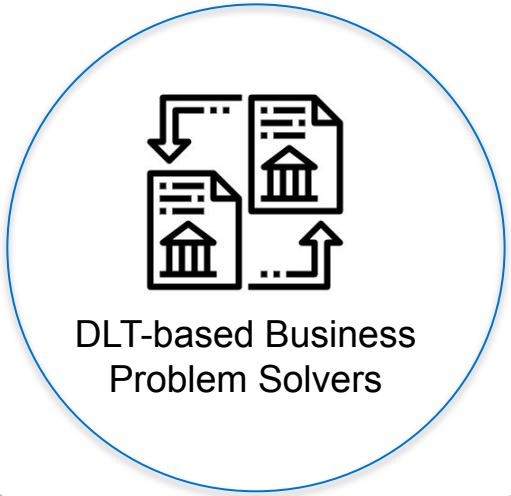
Results
- Feature Catalog
  - Functional Features
  - Non-Functional Features
- Statistical Evaluation of the Use-Cases
  - Feature Usage
  - Platform Distribution
- Evaluation of the Reasonableness of DLT Implementations
- Future Work and Conclusion

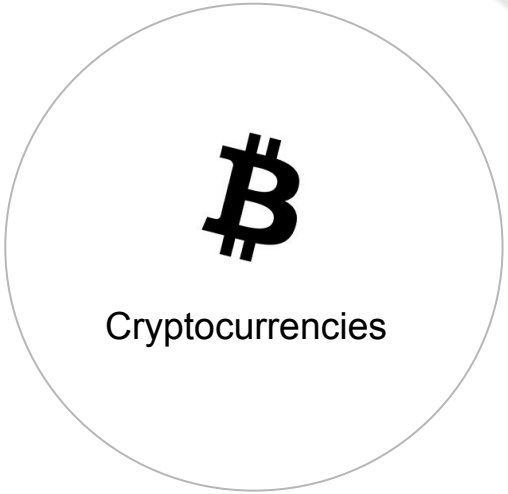
Research Approach – Overview



Icons from Flaticon.com: mavadee srip Good Ware Freepik

Research Approach – Overview





Icons from Flaticon.com: surang smalllikeart

Research Approach – Research Questions

RQ1	What are common features that Distributed Ledger Applications share?
RQ2	How can these features be categorized and formalized?
RQ3	Which decisions are made in the current DLT developments?
RQ4	How can a feature summary be verified?
RQ5	How can the reasonableness of using DLT in projects be evaluated?

Outline

Motivation
- Evolution of DLT
- Academic Publications

Research Approach
- Overview
- Research Questions

Results
- Feature Catalog
  - Functional Features
  - Non-Functional Features
- Statistical Evaluation of the Use-Cases
  - Feature Usage
  - Platform Distribution
- Evaluation of the Reasonableness of DLT Implementations
- Future Work and Conclusion

RQ1: What are common features that Distributed Ledger Technology applications share?

RQ2: How can these features be categorized and formalized?

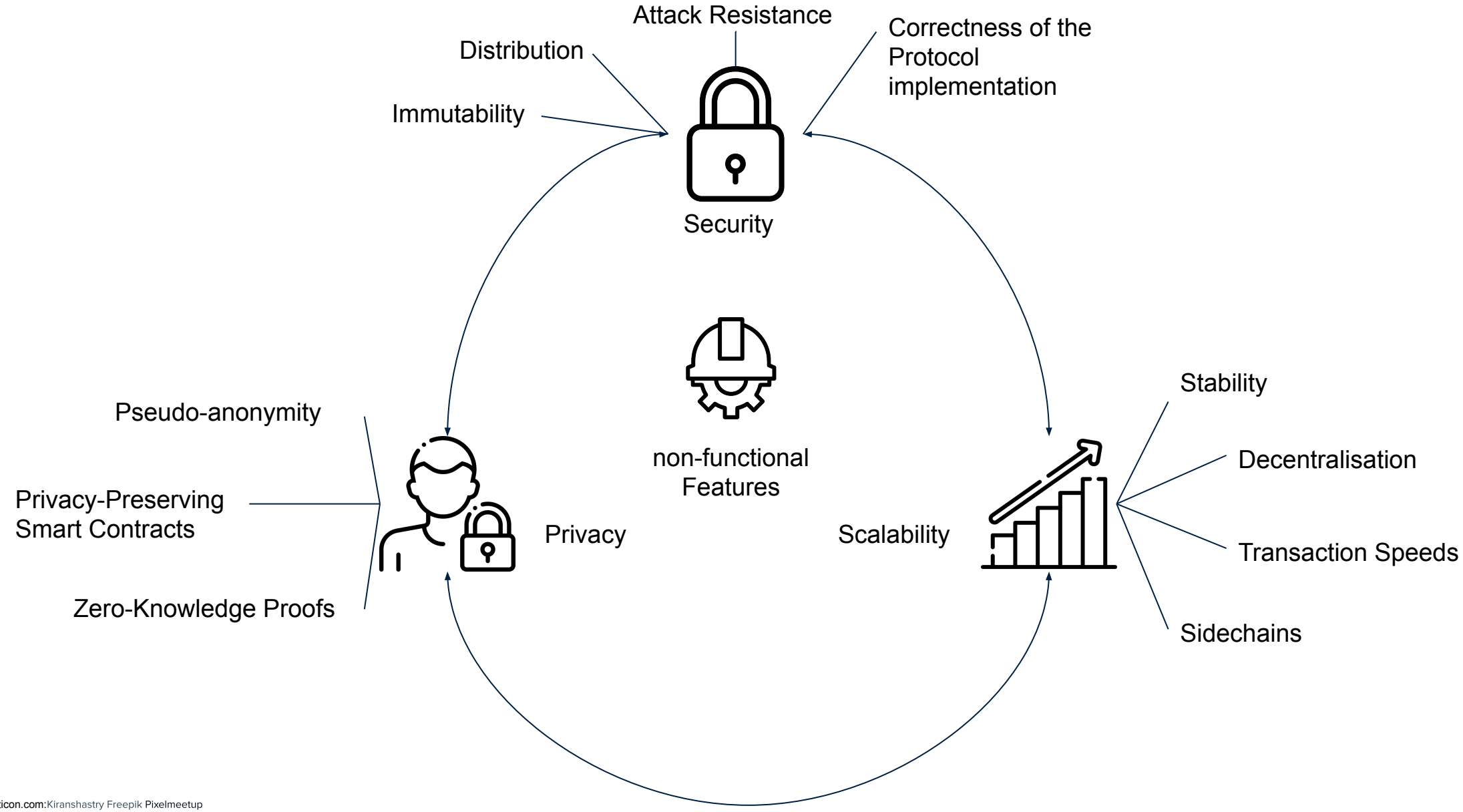
Identity Management	Unique Asset Transfer	Unique Asset Management	Code Execution on the Blockchain
Information Broadcasting	Storing verifiable Data Off-Chain: Hash Anchoring Pattern	Storing Data On-Chain and its Usage	Interoperability
Governance	Cryptocurrency Minting	Cryptocurrency Storing	Cryptocurrency Transfer

RQ1 + RQ2: Selected Functional Features in their Categories

Identity Management	Interoperability	Storing Data On-Chain and its Usage	Code Execution on the Blockchain
• Asymmetric Cryptography • Certificate based Authentication • Self-Sovereign Identities (SSI)	• Atomic Cross-Chain Swaps • Flash swaps, Liquidity pools	• **Encrypting Data on-chain Pattern** • **Oracles Pattern** • **Reverse Oracles Pattern** • Findable Data	• **Execute-Order-Validate Architecture Pattern** • **EVM Architecture Pattern**

Governance	Cryptocurrency Storing	Cryptocurrency Transfer	Cryptocurrency Minting
• Superblocks • Signalling and Voting	• **UTXO Pattern** • **Account Pattern** • Hybrid Approaches	• **Confirmation Pattern** • Single Signature Send • **Multiple Signature Pattern**	• Pre-Mined Cryptocurrency • **Mining Pattern: Proof-Of-Work** • **Staking Pattern: Proof-Of-Stake** • Minting • Elastic Supply

RQ1 + RQ2: Non-functional Features



Icons from Flaticon.com: Kiranshastry Freepik Pixelmeetup

RQ3: Which decisions are made in current DLT developments?

Which problems are the projects that are currently in development solving?

Cross-Organisational Data Storage to enable Use-Cases like Product and Supply Chain Tracing

- BMW - Part Chain is about traceability in the whole supply chain of BMW
- IBO - Recording Quality-Documentation and making it available

Proof-Of-Origin, Ownership or Authenticity of Assets

- CodeNotary - Create Trust in Digital Objects
- Lakoma - Proof of sustainable production across company borders
- FfE - Blockchain-based proof of origin for electricity in a high resolution
- Bernstein - Secure the Ownership of Intellectual Property

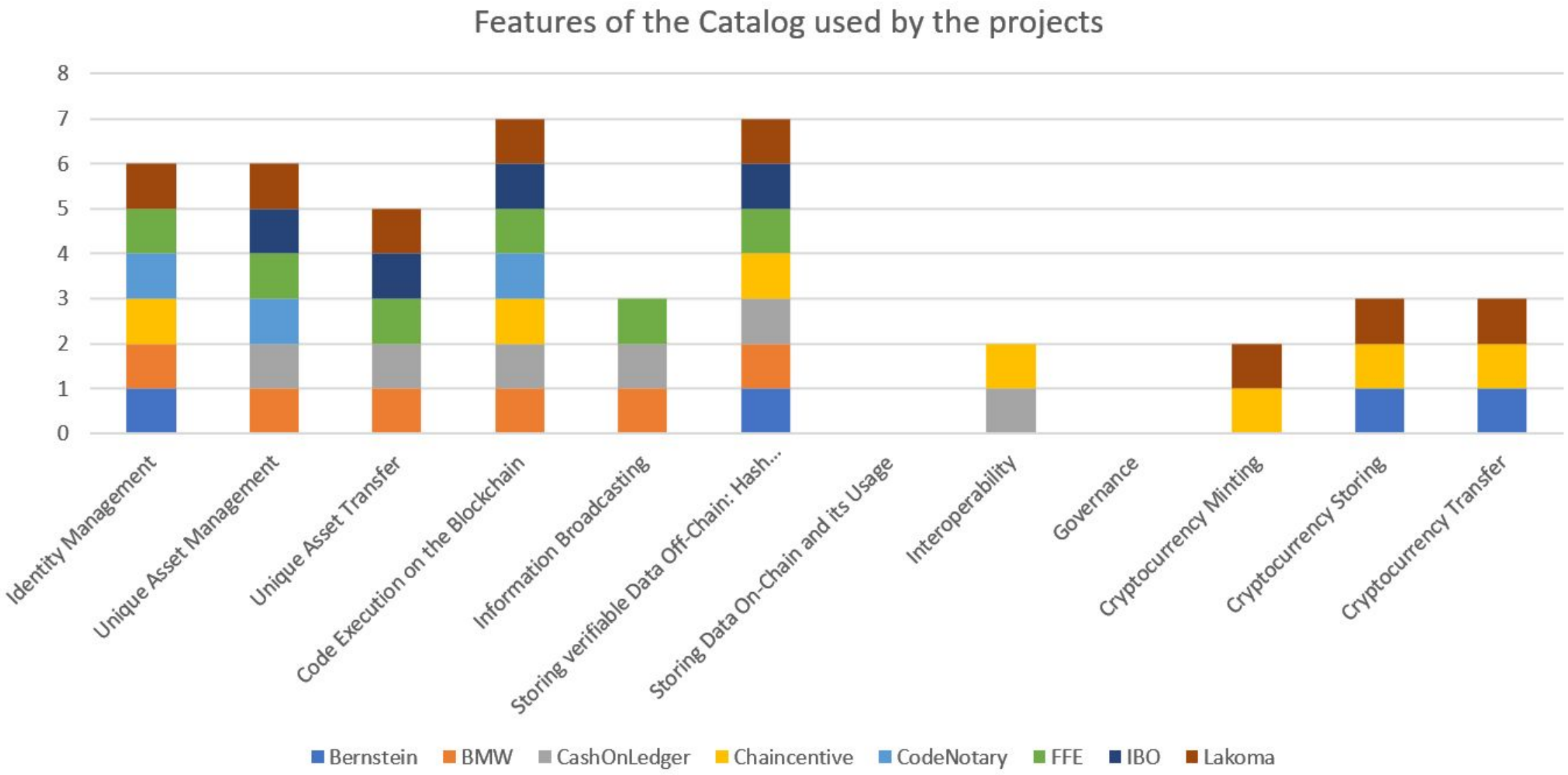
Payment Processing

- Cash on Ledger - Orchestration of payments using Blockchain-Technology
- Chaincentive - Incentives for the positive change in behavior

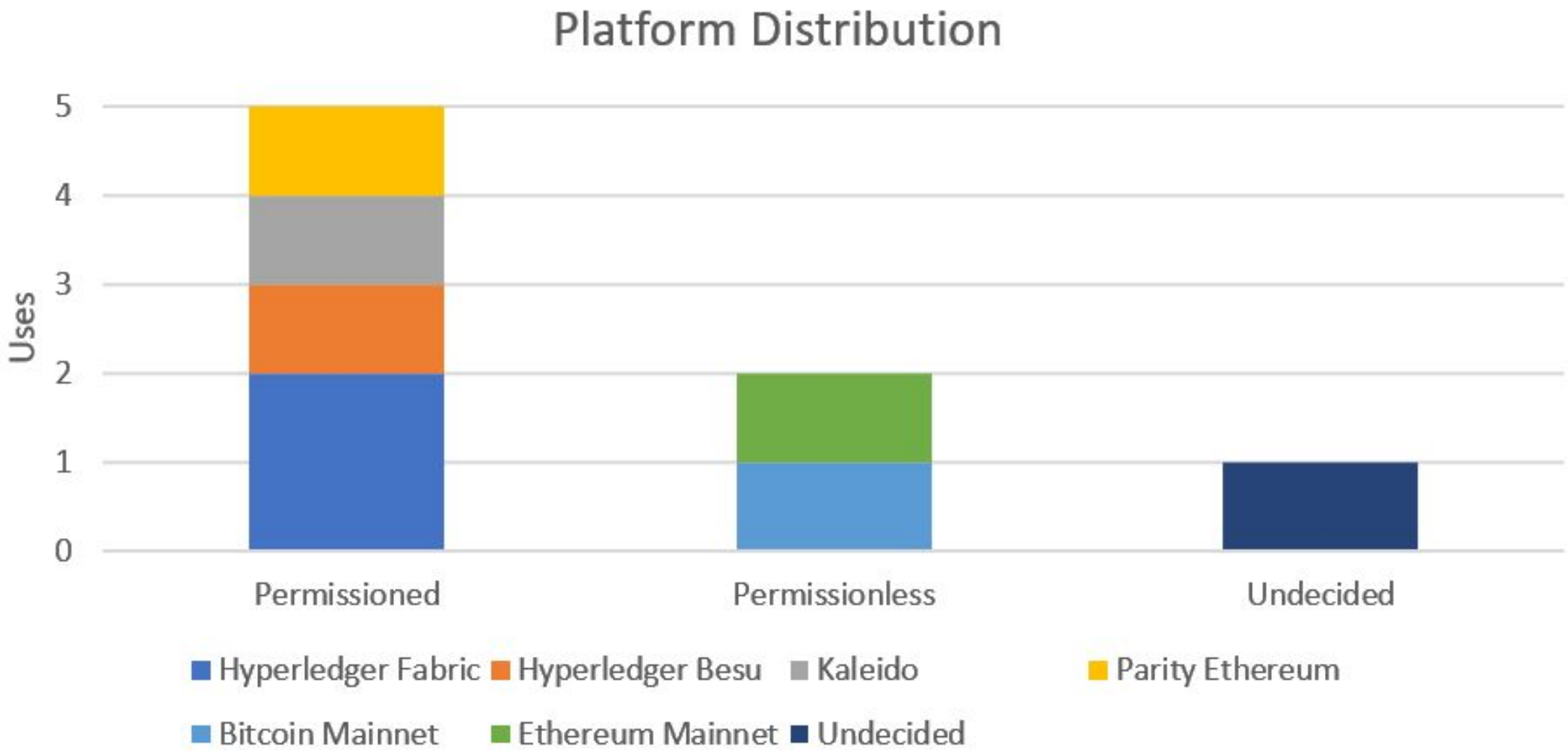
What are re-occurring questions in the architecture of enterprise DLT applications?

- permissioned or permissionless platform
- specific platform of the category permissioned, permissionless, or the decision to create an own platform
- frameworks and design patterns within the platform of choice

RQ4: How can a feature summary be verified?



Results – Statistical Evaluation of the Use-Cases



RQ5: What are characteristics for proper DLT applications?

Why is DLT needed for solving the specific issues?

- Elimination of third parties for profit
- The Blockchain is uncensorable
- Sharing data and their business benefits across company borders
- Many actors with different needs on a single platform can securely interact with each other
- Privacy of the data and transparency of the correctness
- Pseudo-anonymity and digital identities
- Tamper-proof and immutable data storage

Could DLT Projects also be realized without DLT and still solve the same issues?

- Only the use-case of CashOnLedger could be solved without DLT
- All others require DLT because of Immutability, Trust-Issues etc.

Would it be possible that projects are easier to implement and execute without DLT?

- No project is easier to implement without DLT in our Use-Case Study
- DLT is used to solve problems more effectively

Results – Future Work and Conclusion

Future Work

- Expand the existing feature summary to a pattern catalog
- Find forces, known uses and consequences for each feature to transform it into a pattern
- Create a diagram where each pattern can be positioned to enable a clustering of the pattern
- Create a radar chart for each pattern to describe its position in the DLT spectrum
- Create a evaluation sheet or similar to evaluate DLT use-cases for their justifiability

Conclusion

- It is possible to partition DLT by *features* instead of *permissioned* and *permissionless*
- A lot more use-case studies have to be conducted in order to create a complete pattern catalog
- Non-functional features are driven by forces and can not just be iteratively improved
- The feature catalog is a first step in partitioning the DLT space
- This work could have been improved by conducting *a lot* more case studies

B.Sc.
Christian Ziegler
christian.ziegler@tum.de

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

matthes@in.tum.de
wwwmatthes.in.tum.de