SPLC'22, GRAZ, AUSTRIA

JOHANNES KEPLER UNIVERSITY LINZ

ISSE

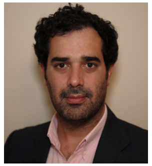
# Variability Debt in Opportunistic Reuse of Existing Products: Literature analysis and a field study

**TD4ViS workshop**
**13 September 2022**

JOHANNES KEPLER UNIVERSITY LINZ

**Wesley Assunção**
ISSE - JKU
Austria

tecnalia
MEMBER OF BASQUE RESEARCH & TECHNOLOGY ALLIANCE



**Jabier Martinez**
Tecnalia
Spain

BIOPARK

**Daniele Wolfart**
Biopark Education
Brazil

# Variability management

- Numerous design and implementation decisions…
  - a relevant source of **technical debt**
- Variability-intensive systems might accumulate **specific types** of technical debt
  - requirements
  - architecture
  - source code
  - documentation
  - tests

# “Variability debt”

- … a type of technical debt caused by issues and sub-optimal solutions in the implementation of variability management in software systems

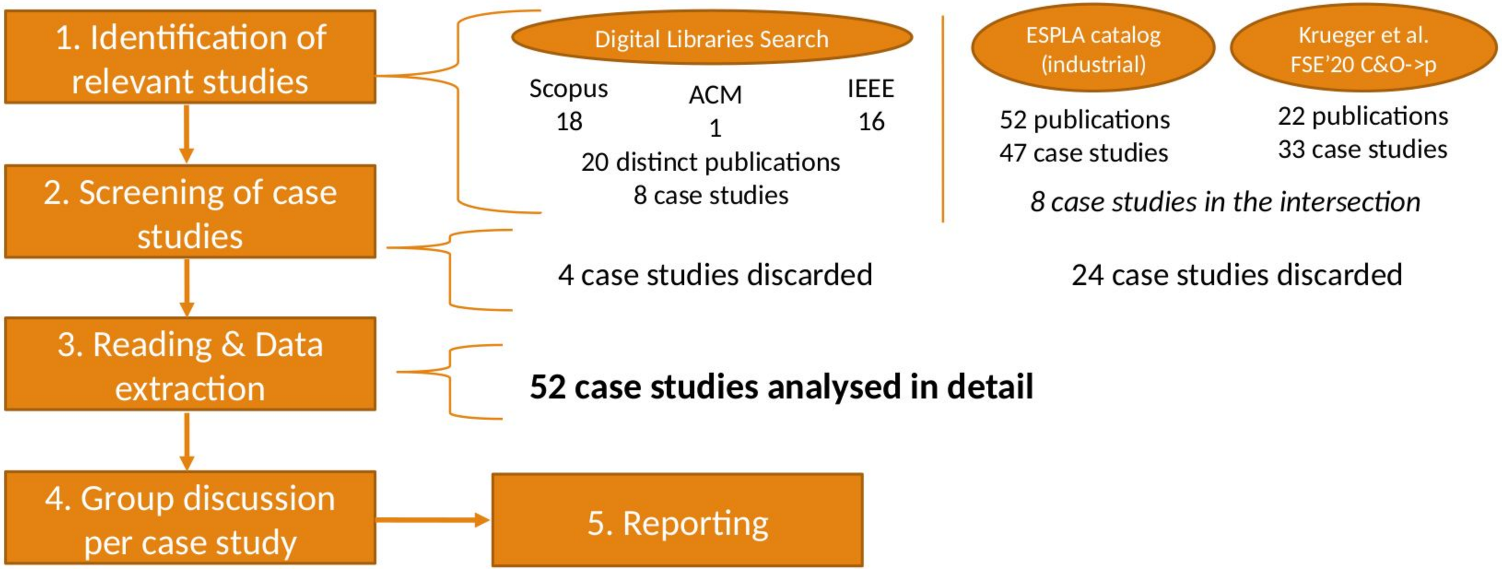
# Variability debt

- We started with an obvious case
  - Clone-and-own opportunistic reuse and the technical debt it represents for the maintenance and evolution of a family of systems as a whole.

*Companies may choose to implement variability with an approach that would bring short-term benefits, for example, using opportunistic reuse, regardless of medium to long-term technical/architectural disadvantages. The debt of variability leads to maintenance and evolution difficulties to manage families of systems or highly configurable systems.*

# Literature review

- 52 industrial cases of variants created through clone-and-own
- Results:
    - causes
    - impacted assets
    - consequences
    - catalog of variability debt items



1. Identification of relevant studies
2. Screening of case studies
3. Reading & Data extraction
4. Group discussion per case study
5. Reporting
Digital Libraries Search
Scopus 18
ACM 1
IEEE 16
20 distinct publications
8 case studies
ESPLA catalog (industrial)
52 publications
47 case studies
Krueger et al. FSE'20 C&O->p
22 publications
33 case studies
*8 case studies in the intersection*
4 case studies discarded
24 case studies discarded
**52 case studies analysed in detail**

Daniele Wolfart, Wesley K. G. Assunção, and Jabier Martinez. "Variability Debt: Characterization, Causes and Consequences." *XX Brazilian Symposium on Software Quality*. 2021.

# Literature review: Results

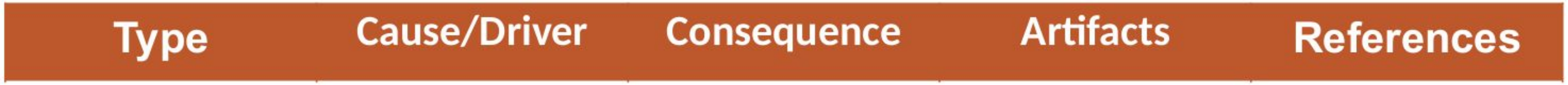
| Causes/Drivers | # Case studies |
|---|---|
| Time pressure | 33 |
| Constantly changing scenarios | 28 |
| Lack of knowledge on variability management | 20 |
| Operational constraints | 7 |

| Assets | # Case studies |
|---|---|
| Source code | 44 |
| Requirements | 30 |
| Architecture | 28 |
| Models | 25 |
| Test cases | 17 |
| DB schemas | 1 |
| Build scripts | 1 |

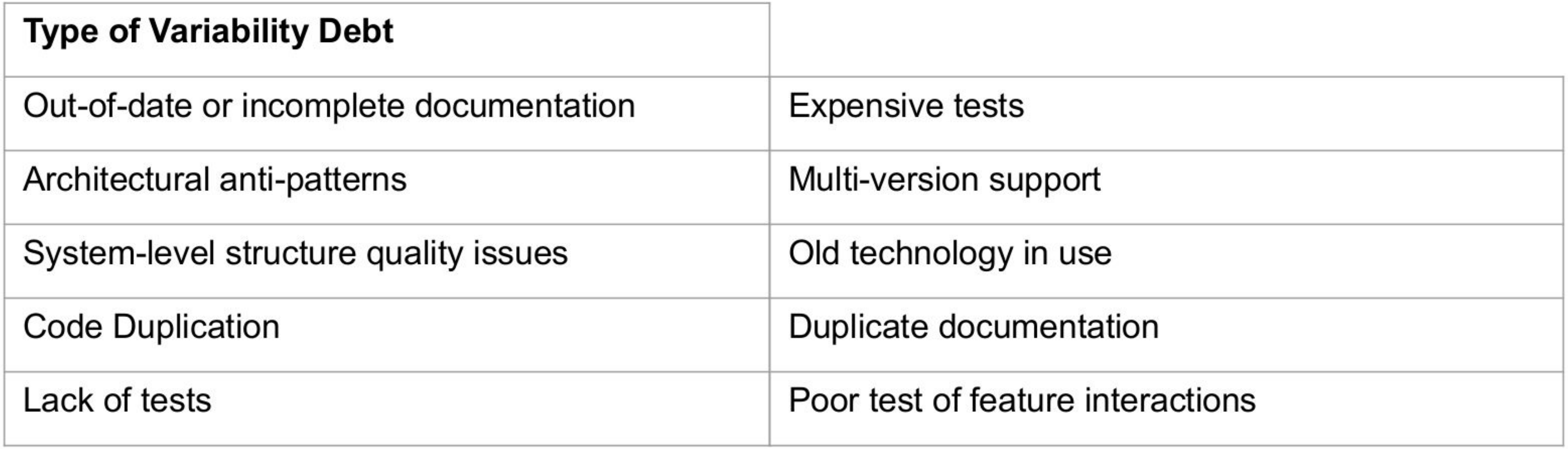
| Consequences | # Case studies |
|---|---|
| Complex maintenance of independent variants | 32 |
| Inability to systematically deal with customization | 28 |
| Inability to create complex systems | 20 |
| Poor overall internal quality | 20 |
| Complex test cycles | 15 |
| Creating usability problems | 3 |
| Portfolio management problems | 3 |
| Repeated roles in similar projects | 1 |

Daniele Wolfart, Wesley K. G. Assunção, and Jabier Martinez. "Variability Debt: Characterization, Causes and Consequences." *XX Brazilian Symposium on Software Quality*. 2021.

# Literature review: Results



| Type | Cause/Driver | Consequence | Artifacts | References |
|---|---|---|---|---|

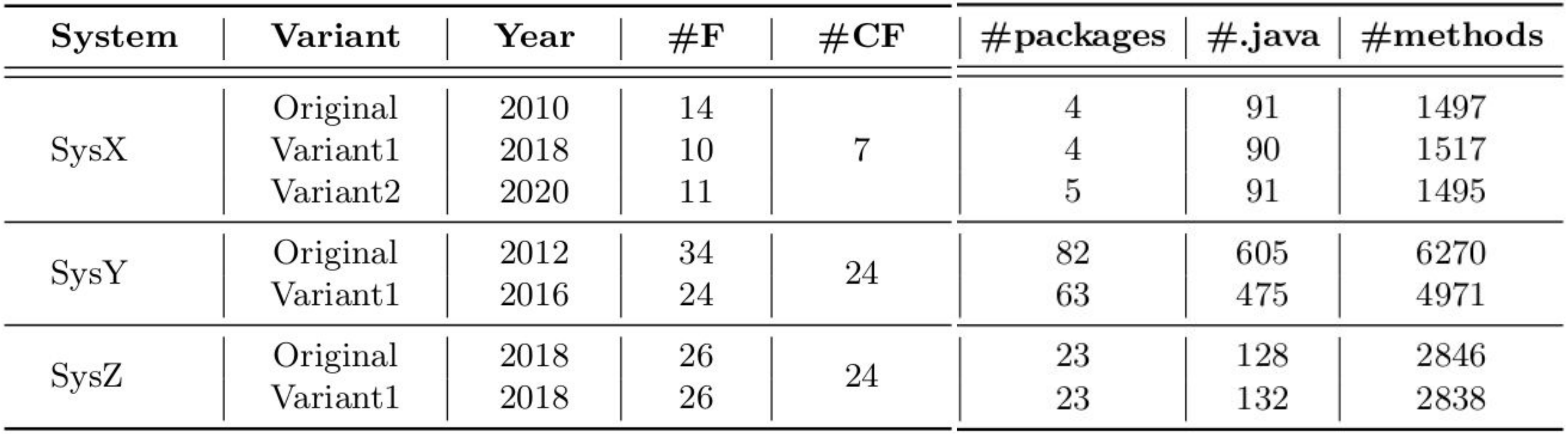


| Type of Variability Debt | |
|---|---|
| Out-of-date or incomplete documentation | Expensive tests |
| Architectural anti-patterns | Multi-version support |
| System-level structure quality issues | Old technology in use |
| Code Duplication | Duplicate documentation |
| Lack of tests | Poor test of feature interactions |

Daniele Wolfart, Wesley K. G. Assunção, and Jabier Martinez. "Variability Debt: Characterization, Causes and Consequences." *XX Brazilian Symposium on Software Quality*. 2021.

# Industrial field study

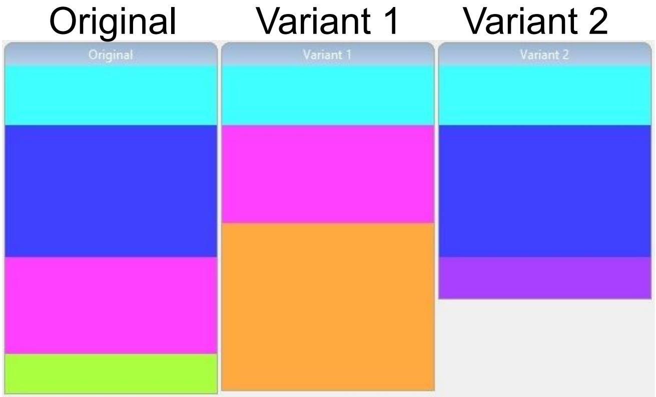
- Variability debt incurred in a real scenario
- Brazilian multinational industry leader in its market segment in Latin America
- Three families of software systems



| System | Variant | Year | #F | #CF | #packages | #.java | #methods |
|---|---|---|---|---|---|---|---|
| SysX | Original | 2010 | 14 | 7 | 4 | 91 | 1497 |
| | Variant1 | 2018 | 10 | | 4 | 90 | 1517 |
| | Variant2 | 2020 | 11 | | 5 | 91 | 1495 |
| SysY | Original | 2012 | 34 | 24 | 82 | 605 | 6270 |
| | Variant1 | 2016 | 24 | | 63 | 475 | 4971 |
| SysZ | Original | 2018 | 26 | 24 | 23 | 128 | 2846 |
| | Variant1 | 2018 | 26 | | 23 | 132 | 2838 |

# Industrial field study: Results

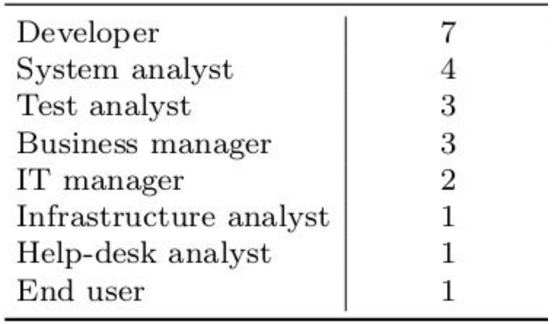
- Evidences of variability debt
  - Variability debt happens in practice
  - Source code is the artifact more copy-pasted for implementing variability in clone-and-own
  - Specifications and Requirements are also replicated for software variants
  - Issues:
    - Replication of business rules
    - Difficult propagation of changes



# Industrial field study: Results

- Survey with stakeholders (22 participants)
  - Managers do not perceive variability debt
  - Technical practitioners clearly agree with the variability debt existence
  - Practitioners mentioned variability debt described in the literature
    - system-level structure quality issues
    - lack of testing
    - multi-version support
    - duplicate documentation

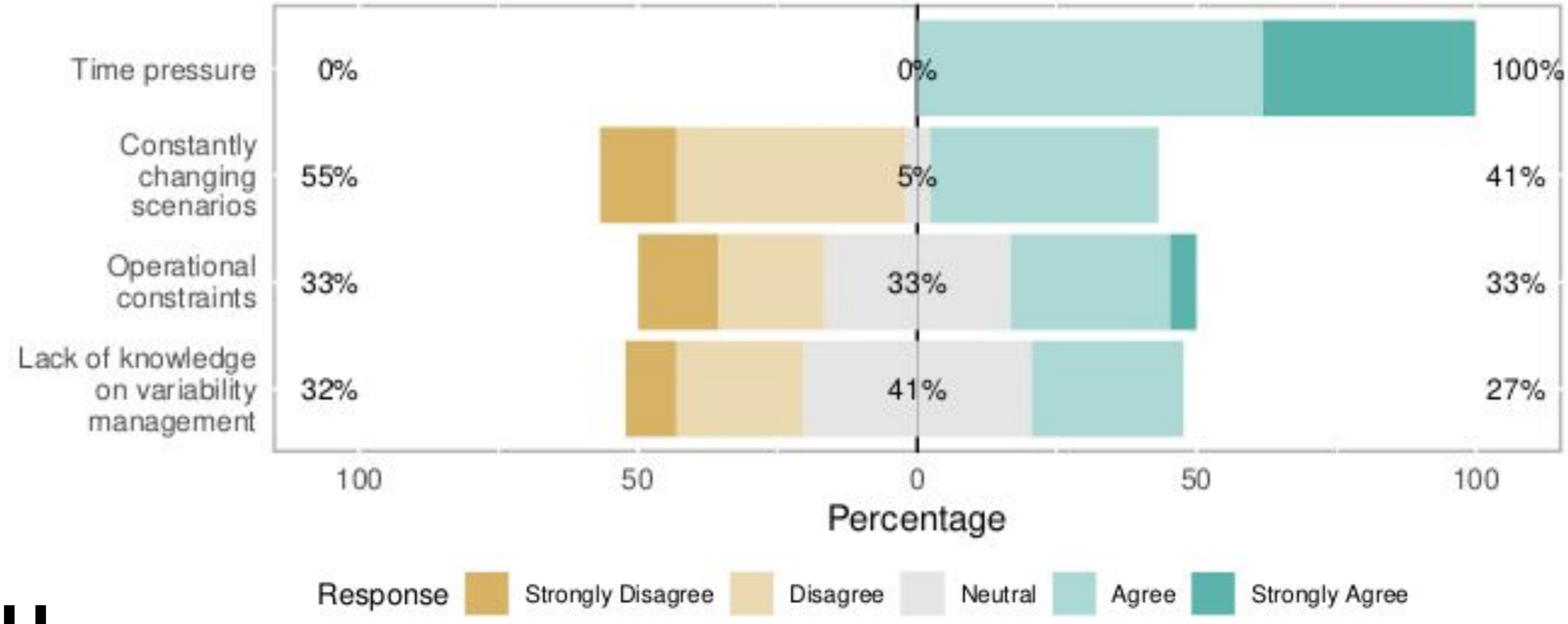
Survey participants



| | |
|---|---|
| Developer | 7 |
| System analyst | 4 |
| Test analyst | 3 |
| Business manager | 3 |
| IT manager | 2 |
| Infrastructure analyst | 1 |
| Help-desk analyst | 1 |
| End user | 1 |

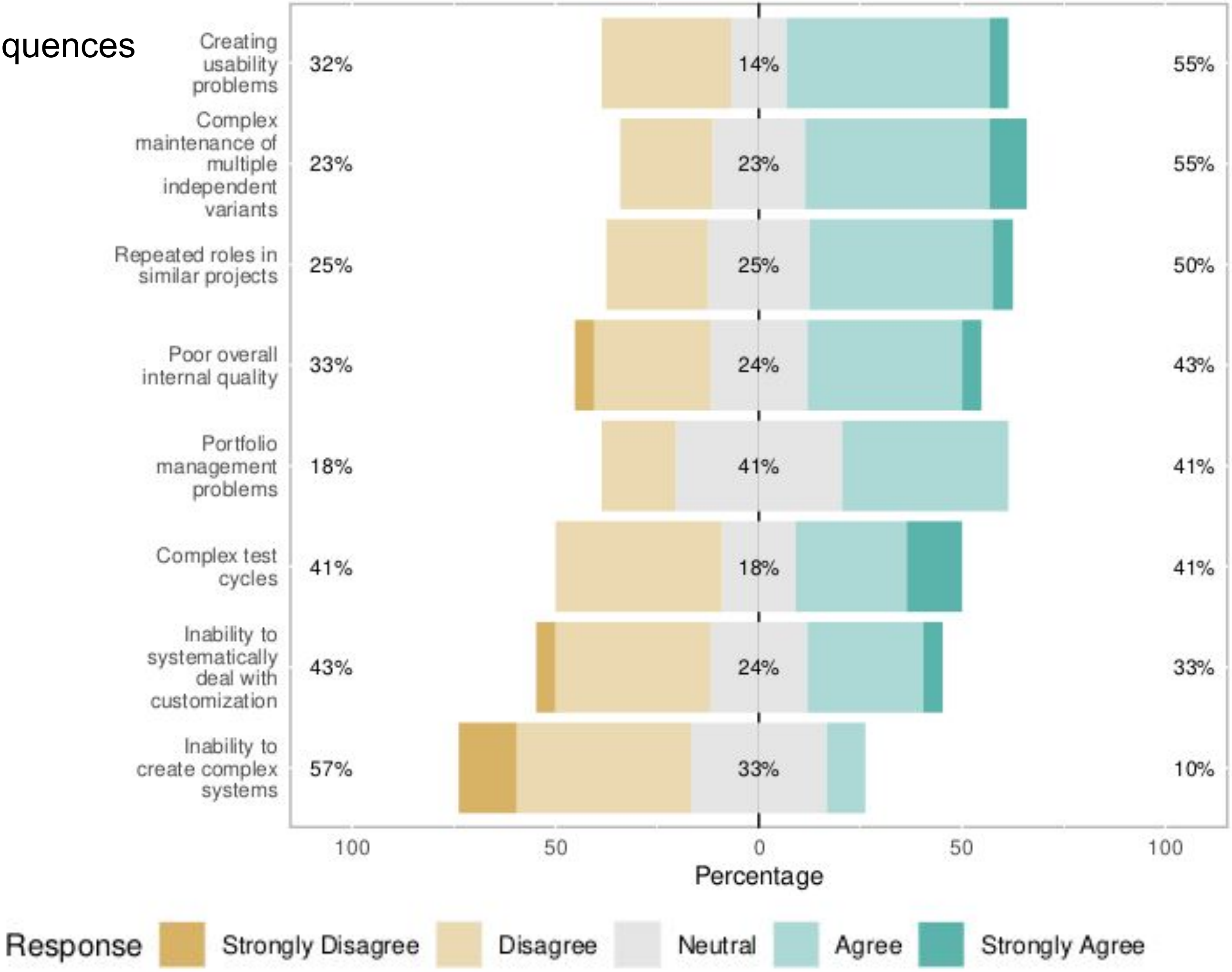
# Industrial field study: Results

Variability debt causes



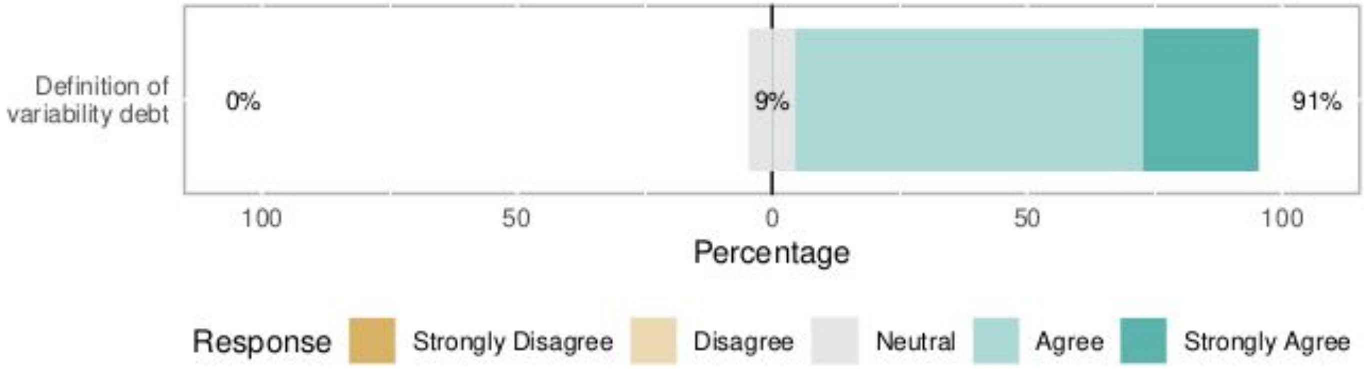
# Industrial field study: Results

Variability debt Consequences



# Industrial field study: Results

Variability debt definition and concept

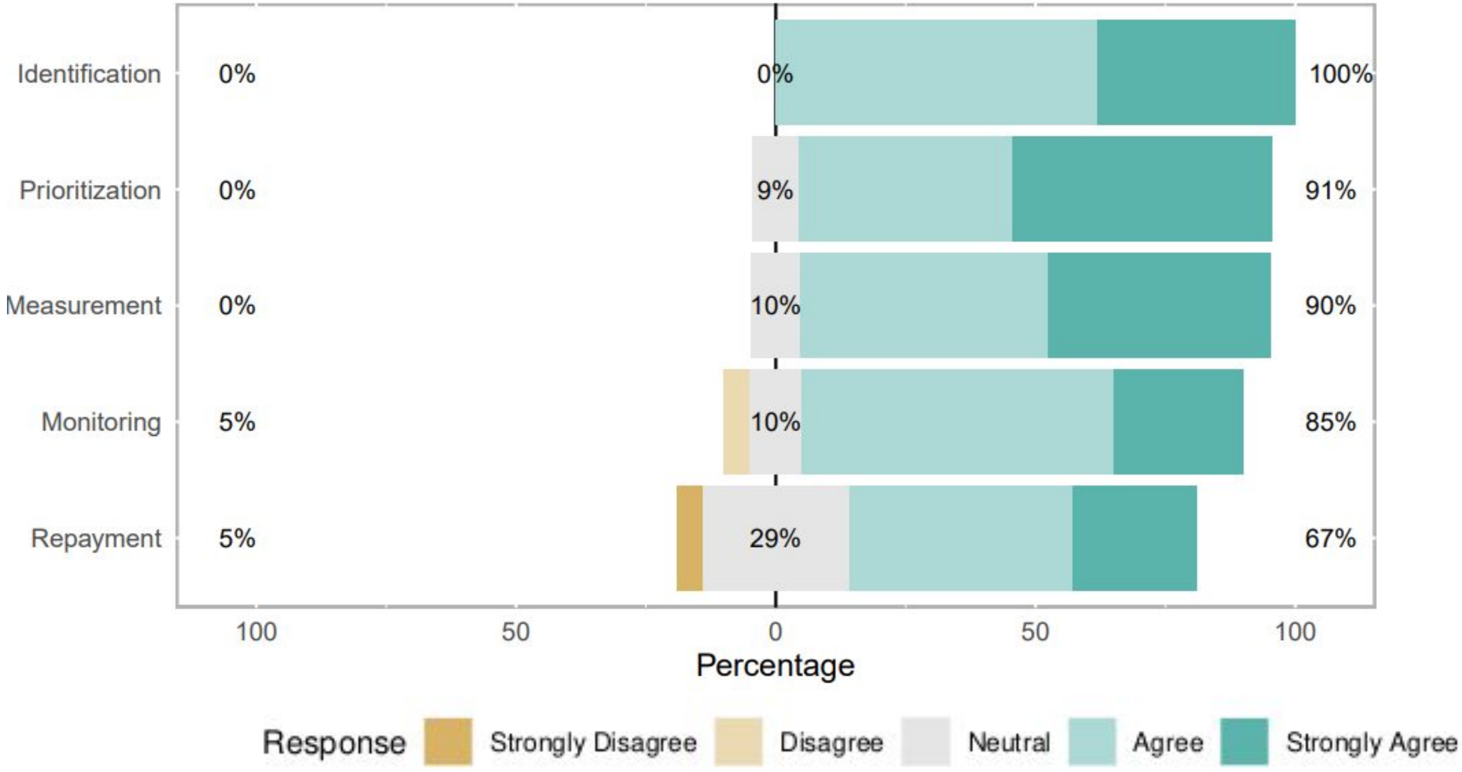


*P1: "I think that the term technical debt is enough. Technical debt occurs for several reasons, and variability is one dimension among them."* ***<- Neutral, but recognizing variability as a reason.***

*P15: "I agree that initially clone-and-own proves to be simpler, but as time goes by, maintaining these architectures becomes much more complex. I also understand that maintaining highly configurable software is extremely complex and that if it is poorly applied, it can cause even greater maintenance or impair system performance"* ***<- Recognizing that variability debt is not only for clone-and-own, but also when you have a running SPL***

# Industrial field study: Results

Perspectives for activities support



# Conclusion

- Both technical debt and variability management are engineering disciplines, with their own roles, methodologies, and practices that have been widely investigated
  - … however, there are only few studies that address them together
- The results of a field study show that variability debt management can be beneficial for the company

# Future directions

- More studies on Variability debt
- Beyond the clone-and-own reuse
  - Software Product Line approaches
  - Feature-oriented approaches
- Characterize variability debt in comparison with other types of technical debt
  - Defining what are the specifics of managing technical debt in variability-intensive systems
- At methodological level, try to merge technical debt management and variability management