

PLEASE2013

4th International Workshop on Product Line Approaches in Software Engineering, May 20, San Francisco, California
held in conjunction with the 35th International Conference on Software Engineering (ICSE 2013)

Requirements-based Delta-oriented SPL Testing

**Michael Dukaczewski, Ina Schaefer,
Remo Lachmann**

**Institute of Software Engineering and
Automotive Informatics**

**Technische Universität Braunschweig
Germany**

Malte Lochau

Real-Time Systems Lab

**Technische Universität Darmstadt
Germany**

Contact Profile

- **Interests:**
 - Model-based software development
 - Modelling of SPL with deltas
 - Testing of variant rich systems
- **Objectives / Looking for**
 - Feedback
 - Industrial case studies

Product Line

- Testing of Software Product Lines based on Requirements
 - Focus: Integration tests
- Can be any SPL where requirements-based testing matters
- Case study in automobile manufacturing industry

The Problem

- It's not possible to test every product variant
- Number of executed test cases have to be reduced
- Which test case should be executed?

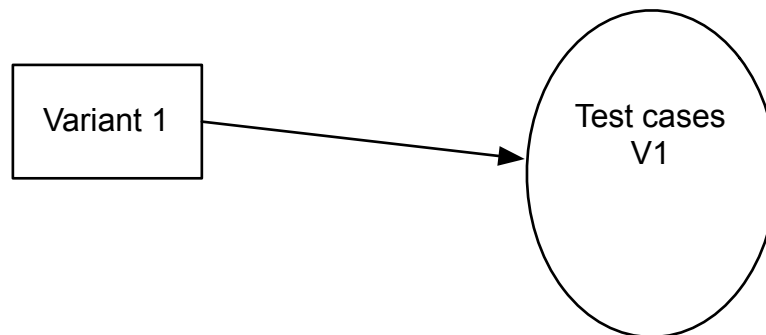


Assumptions

- Set of system requirements
- Test cases for the requirements
- Test cases reference the requirements
- Requirements and test cases in natural language

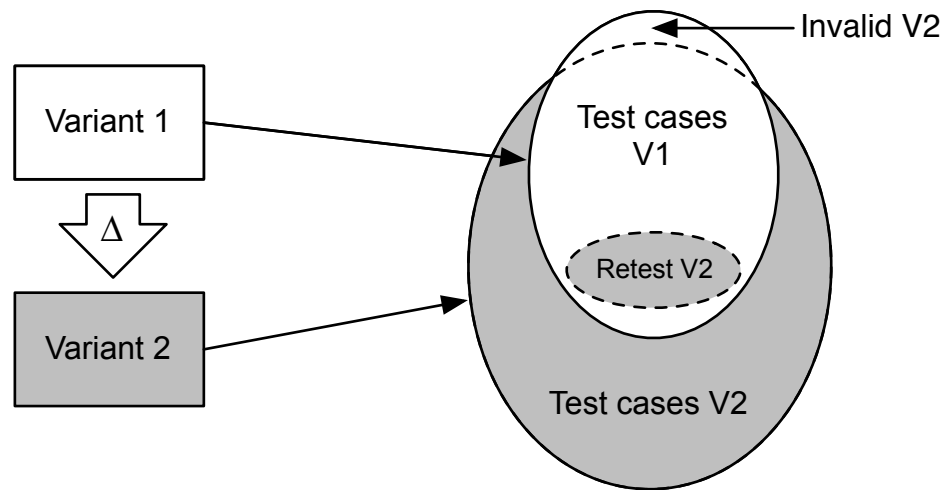
Solution (summary)

- Test one variant fully and then incremental only the changes



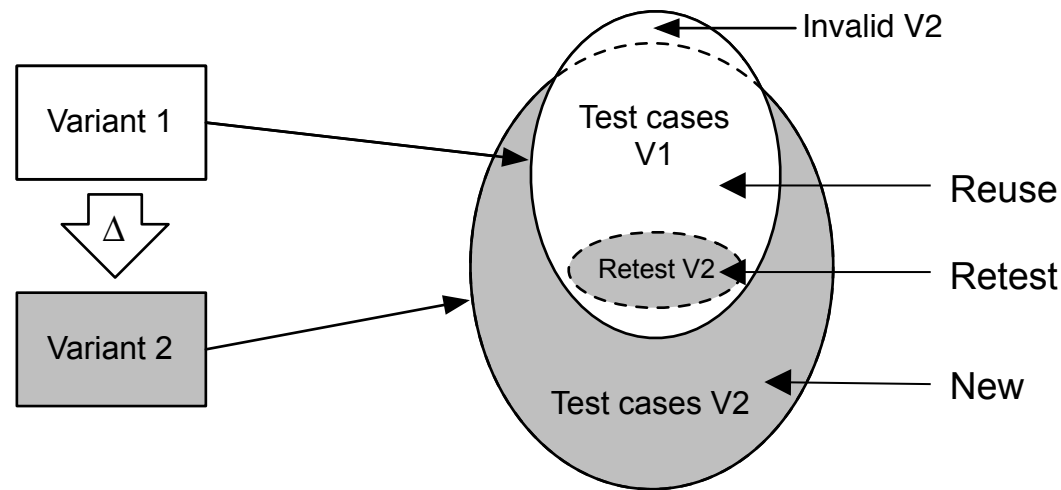
Solution (summary)

- Test one variant fully and then incremental only the changes



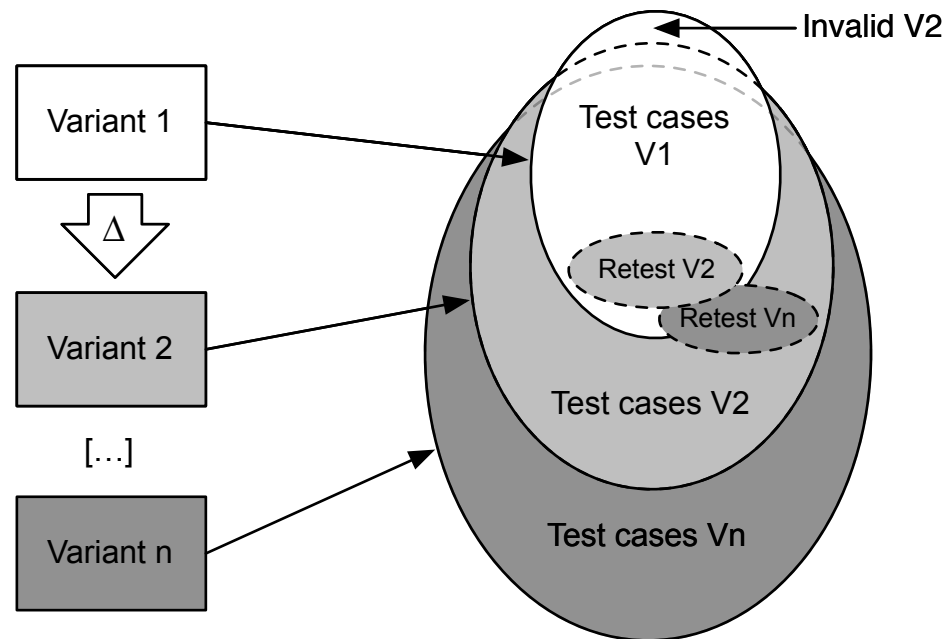
Solution (summary)

- Test one variant fully and then incremental only the changes



Solution (summary)

- Test one variant fully and then incremental only the changes



Example

Requirements

BCS_R1

If an object is detected in the window (window pressure $P >$ threshold), activate the finger protection to prevent the power window from moving any further.

BCS_R2

If the central locking system is activated and the power window is not in the top position, move the power window up, until it reaches the top position.

BCS_R3

If the move down button for the power window is pressed and there is no Central locking system, move the power Window down. Otherwise, only move down if the central locking system is deactivated

BCS_R4

After the move up button has been tapped shortly (< 1 sec), the power window moves automatically up until it reaches the top position and then the movement stops.

...

BCS_Rn

Test cases

BCS_TC1

Precondition: Window is open and an object is within the window
 Action: Press move up button
 Expected Result: Window moves up, until it reaches the objects and stops

BCS_TC2

Precondition: CLS is activated & power window is not in top position
 Action: Press move up button
 Expected Result: Power window moves to the top position and stops

BCS_TC3

Precondition: No CLS installed
 Action: Press move down button
 Expected Result: Power window moves to the bottom position and stops

BCS_TC4

Precondition: CLS installed and deactivated
 Action: Press move down button
 Expected Result: Power window moves to the bottom position and stops

BCS_TC5

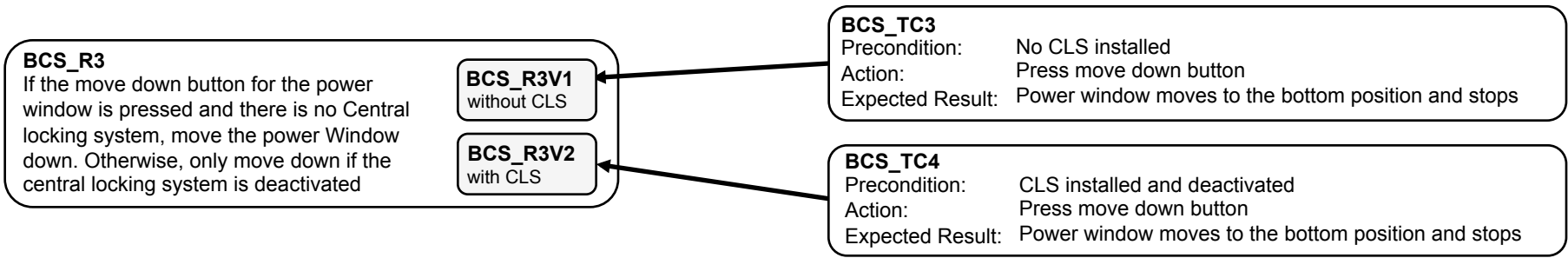
Precondition: Power window is at bottom position
 Action: Press move up button for less then 1 second
 Expected Result: Power window moves to the top position and stops

...

BCS_TCn

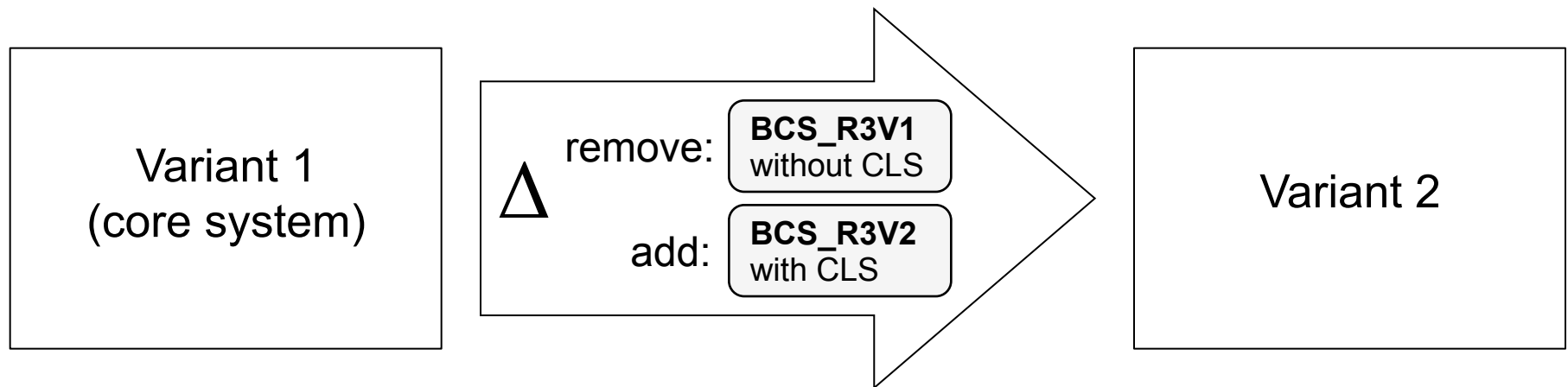
Solution (Step 1)

- Split requirements, so that each cover only one feature



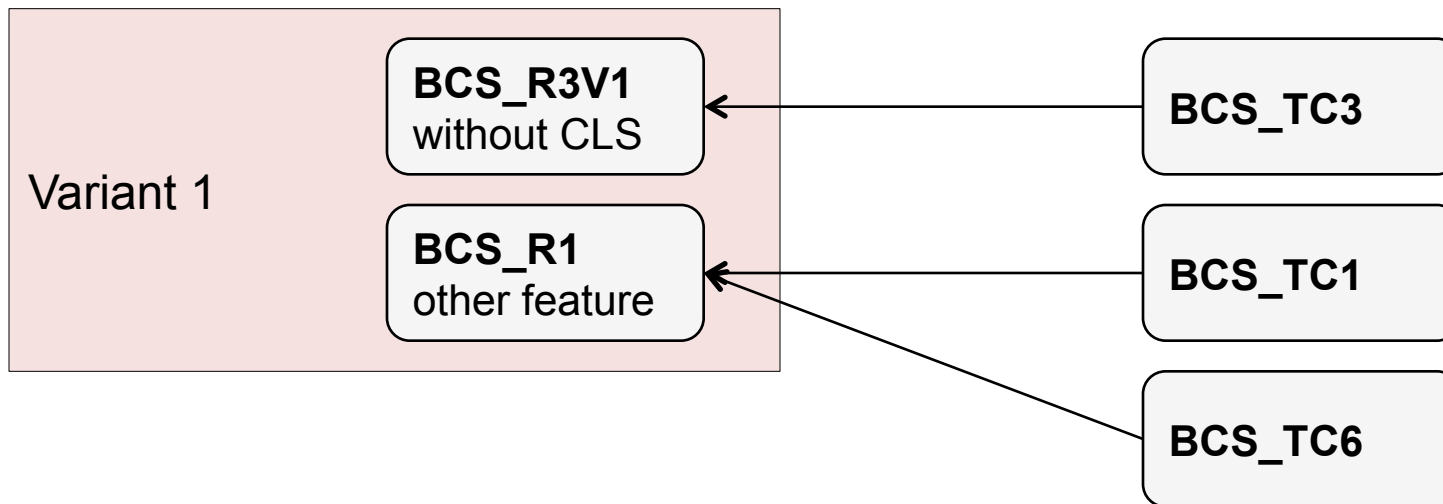
Solution (Step 2)

- Separate requirements into a core system and define deltas for variants
- Apply deltas to get variants



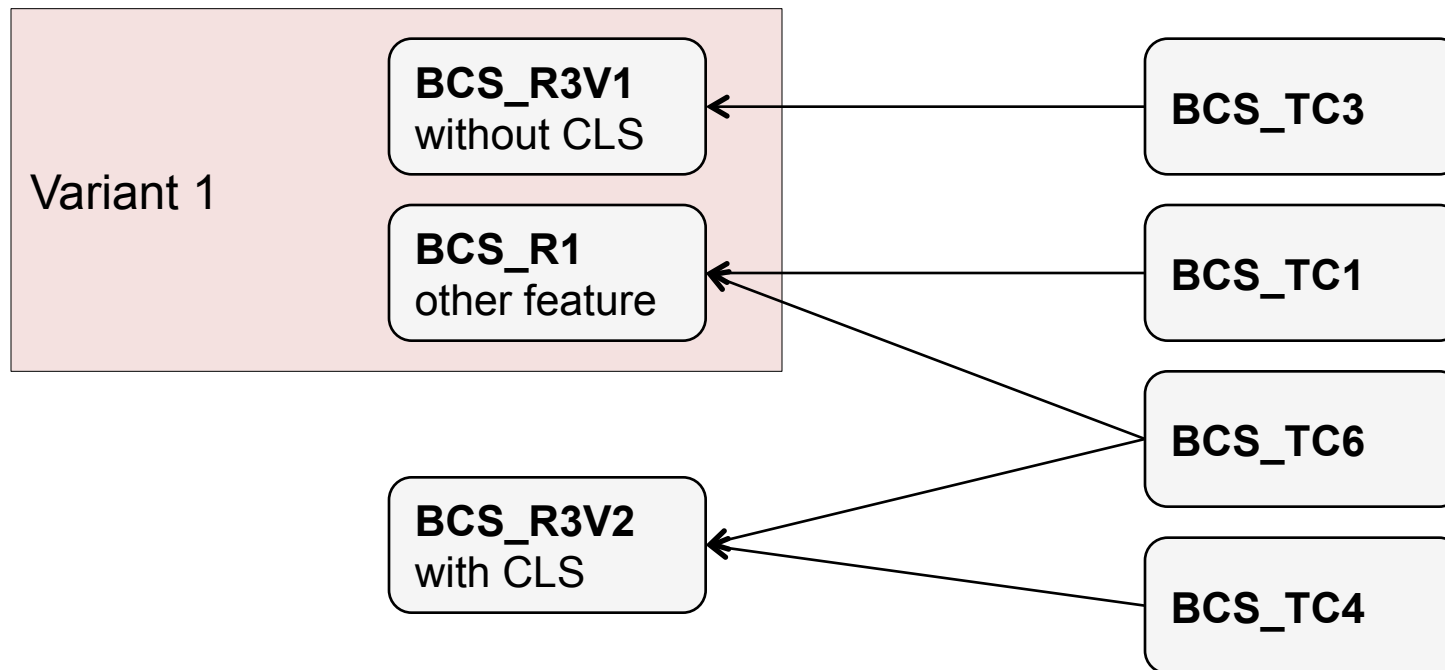
Solution (Step 3)

- Categorize test cases



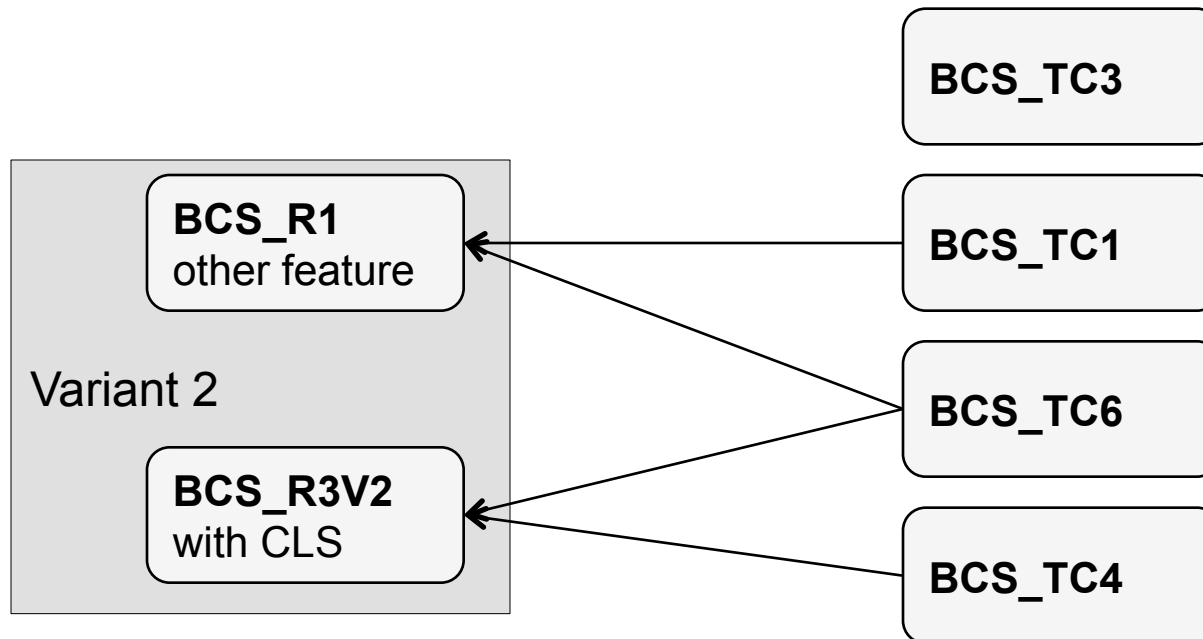
Solution (Step 3)

- Categorize test cases



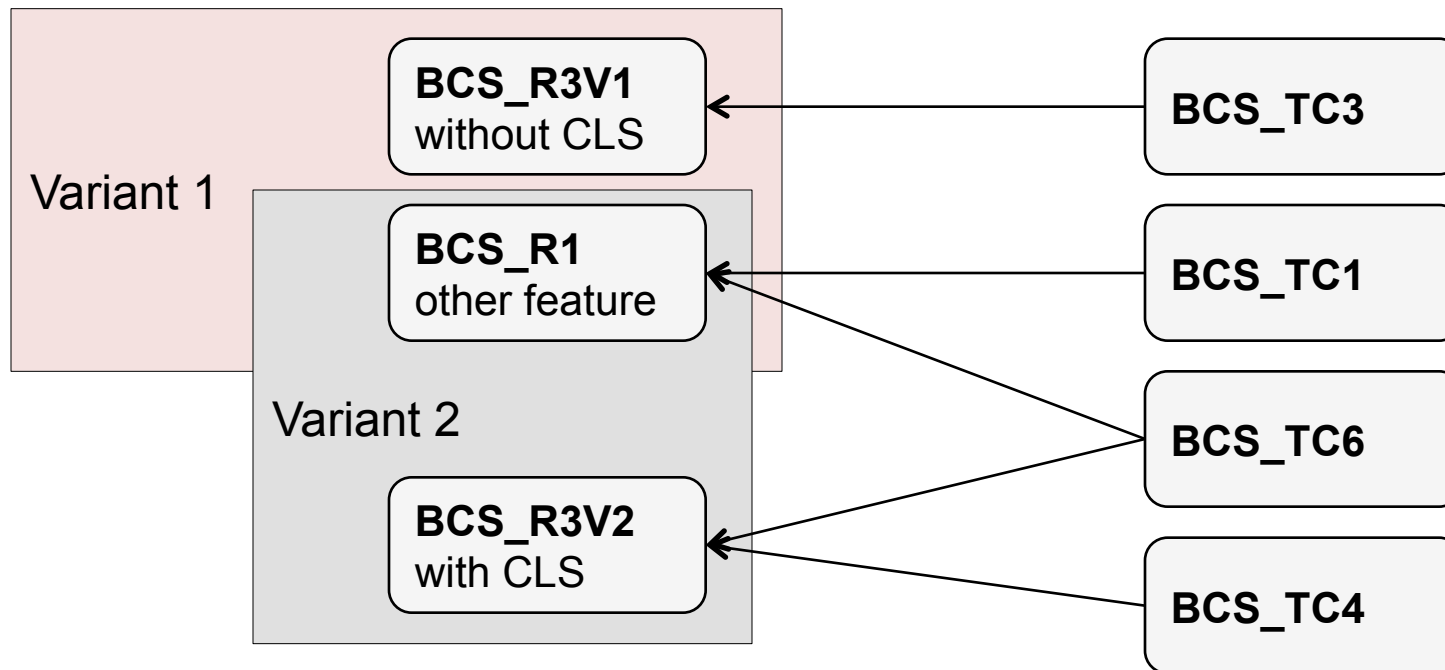
Solution (Step 3)

- Categorize test cases



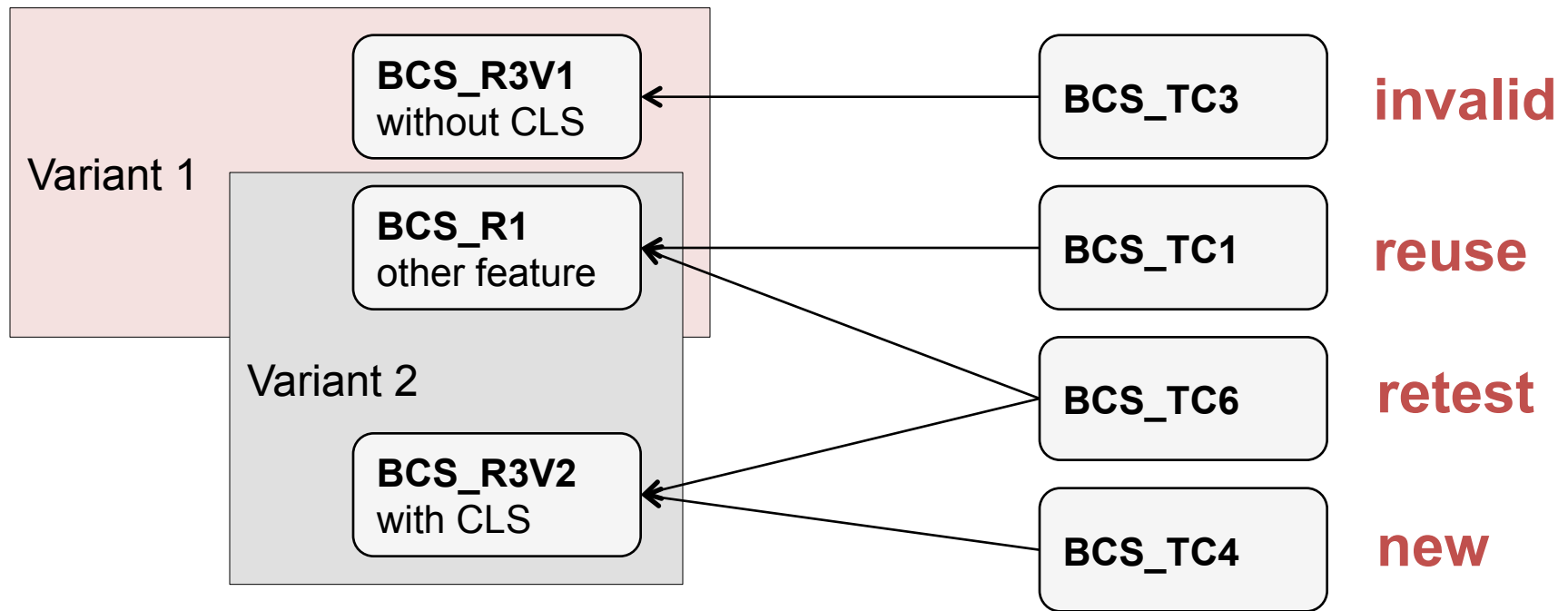
Solution (Step 3)

- Categorize test cases

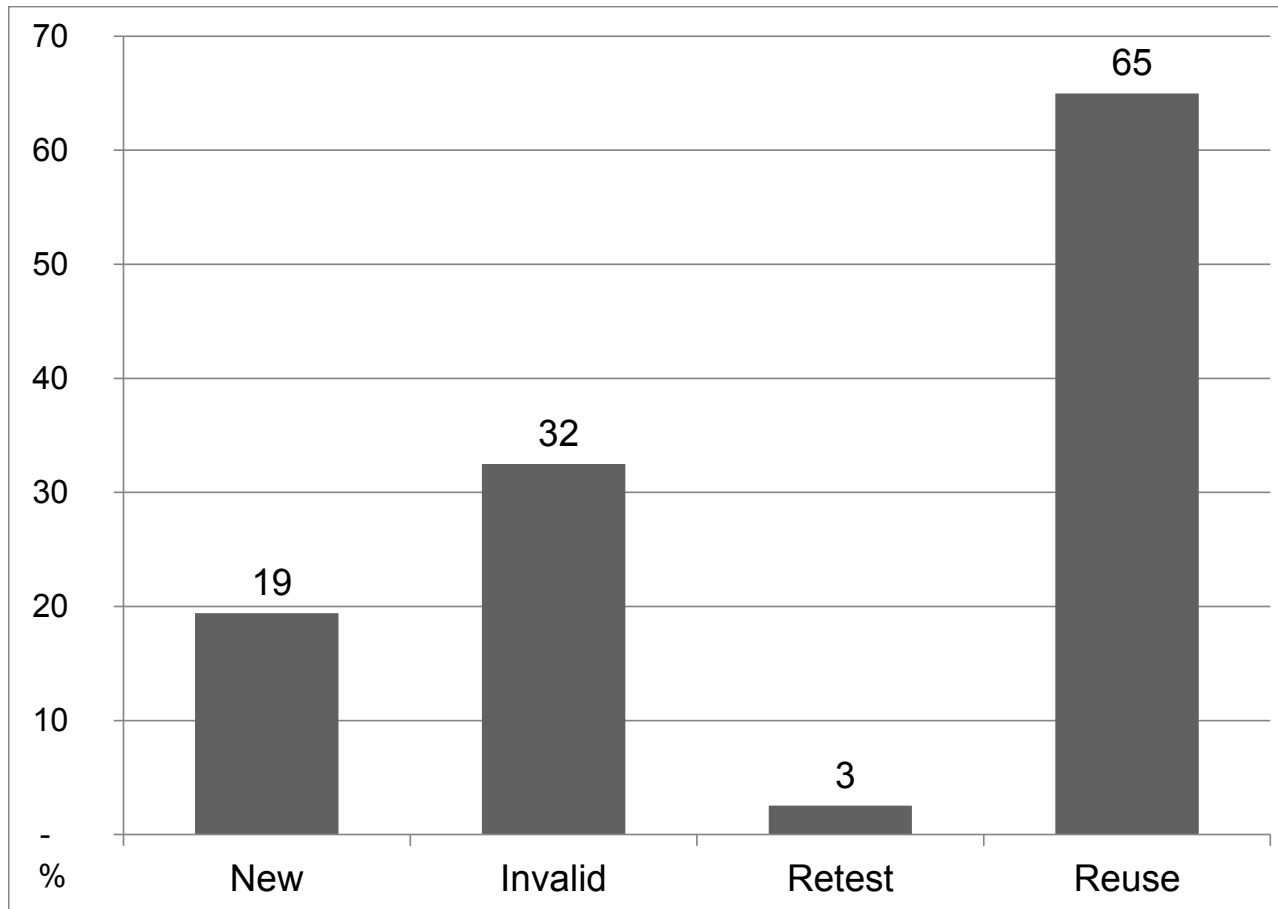


Solution (Step 3)

- Categorize test cases

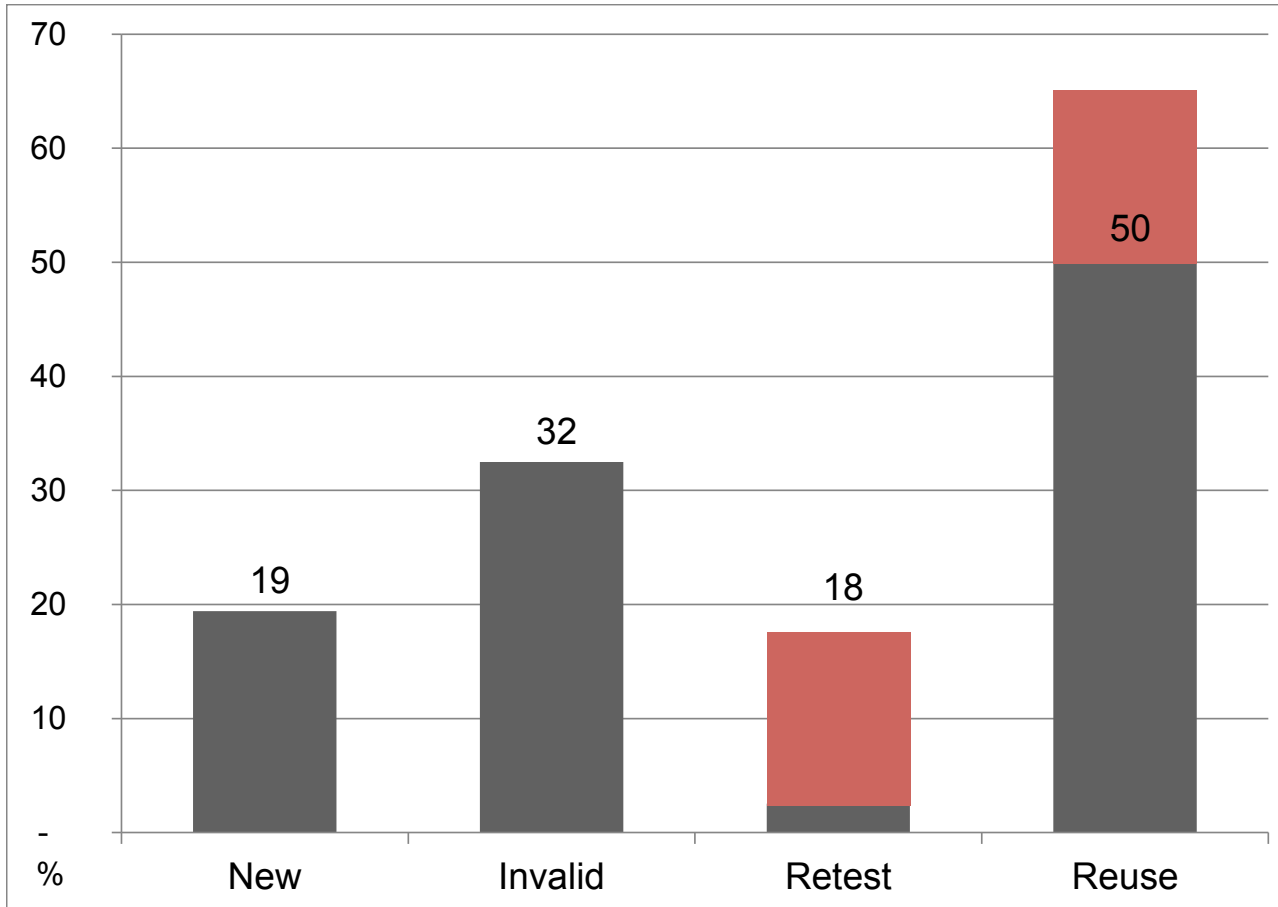


Case Study (Results)



Case study in automotive
manufacturing industry
with 2 Variants

Case Study (Results)



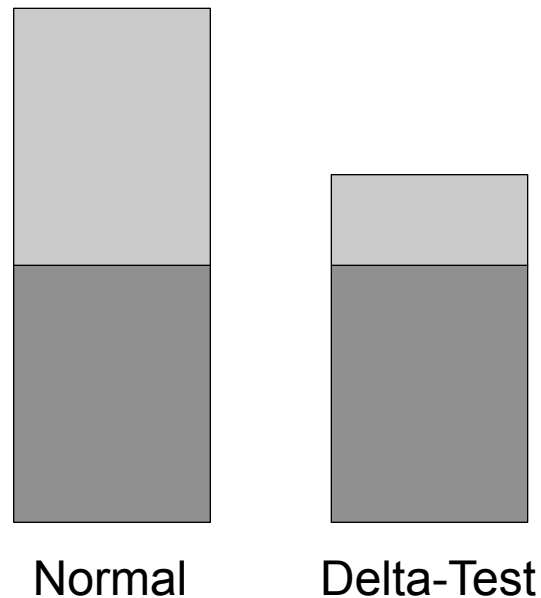
Case study in automotive manufacturing industry with 2 Variants

Variant 1: 100% Invalid + Retest + Reuse

Variant 2: 37% New + Retest

Why is it interesting?

- Testing effort can be reduced by using our delta-based strategy



Discussion

- What one has to do to in order to use our solution?
- How difficult is it to use the solution? Under which conditions can the solution be applied?
- How good is the error detection?

Thank you!

Michael Dukaczewski, TU Braunschweig