



# Problems and Issues

**Jan Bosch**  
*Professor of Software Engineering  
University of Groningen, Netherlands  
Jan@JanBosch.com  
<http://www.janbosch.com>*



## Problems and Issues

- lack of first-class representation of variability mechanisms (relation to why)
- lack of first class representation of dependencies
- lifecycle management of variation point

# Analysis

- stakeholder concept overlap
- overgeneralisation & overtrivialisation
- no awareness wrt binding time

# Design

- architect oversimplifies variability needs (e.g. maps everything to one mechanism)
- lack of awareness of pro/cons/impact of variability mechanisms
- lack of flexibility for flexibility mechanisms
- resource awareness in relation to variability mechanisms

# Coding

- Complexity and size of mapped / transformed variation point (1-to-n mapping) a.k.a. ripple effect a.k.a. SPLAT!

# Run Time

- resource cost (memory cost, performance, etc.)
- lack of predictability of quality attributes and functionality
- higher development resource needs
- certification (e.g. security)
- side-by-side problem
- uncertified 3<sup>rd</sup> party variants problems
- how to sandbox uncertified variants
- lack of explicit representation of context dependencies

# Case Study Results

- ConIPF EU project
- case study projects
  - Robert Bosch GmbH
  - Thales Naval Netherlands
- University of Hamburg

based on

Sybren Deelstra, Marco Sinnema & Jan Bosch, "Product Derivation in Software Product Families - A Case Study", Journal of Systems and Software (to appear)

# Knowledge Externalization

observed problems

- very high workload of experts
- false positives of compatibility check during component selection
- large number of errors in parameter settings due to large amount of parameters with implicit dependencies

# Knowledge Externalization

observed problems (continued)

- identical errors in successive projects due to lack of externalizing important implicit dependencies
- over-explicit documentation decreases traceability of relevant information

# Knowledge Externalization

causes and forces

pro externalization

- avoiding knowledge starvation
- avoiding known errors as a result of implicit dependencies
- enabling tool support

# Knowledge Externalization

causes and forces

con externalization

- externalization process intervenes with every day business
- externalization expensive and time consuming
- investment in infrastructure
- semantics
- maintenance

# Variability Management

observed problems

- unmanageable number of variation points and variants
- variation points not organized hierarchically
- consequences of variant selection unclear
- limited resources for realizing variability

# Variability Management

observed problems (continued)

- testability
- no uniform treatment of variation points over the lifecycle
- variation point and mechanism considered identical

# Variability Management

causes and forces

- variability realization
- managing combinatorial explosion

## Scoping and Evolution

observed problems

- repetition of development
- scoping of features
- road mapping
- obsolete variation points not removed

## Scoping and Evolution

observed problems (continued)

- derivation costs higher than expected due to required adaptations
- combinatorial explosion due to different provided and required interfaces

# Scoping and Evolution

causes and forces

- product specific adaptation versus reactive evolution
  - product specific adaptation
  - reactive evolution

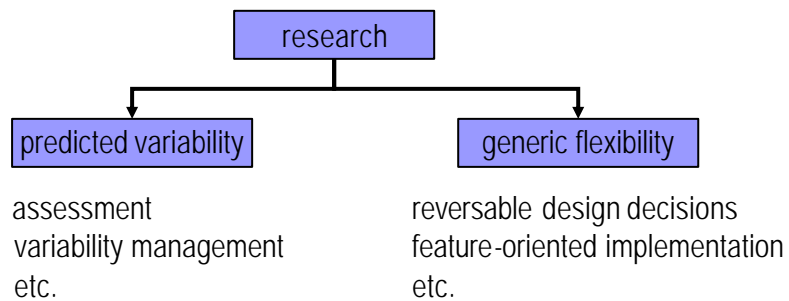
# Scoping and Evolution

causes and forces

- product specific variation points
- addition versus removal

# Outlook

- Robert Bosch GmbH:  
50-60%, 30-40%, 10% problem



# Conclusion

- problems and issues
  - general
  - analysis, design, coding, run-time
  - knowledge externalization
  - variability management
  - scoping and evolution
- outlook