

# Software Clones in Product Lines

Rainer Koschke

Arbeitsgruppe Softwaretechnik  
Fachbereich Mathematik und Informatik  
Universität Bremen

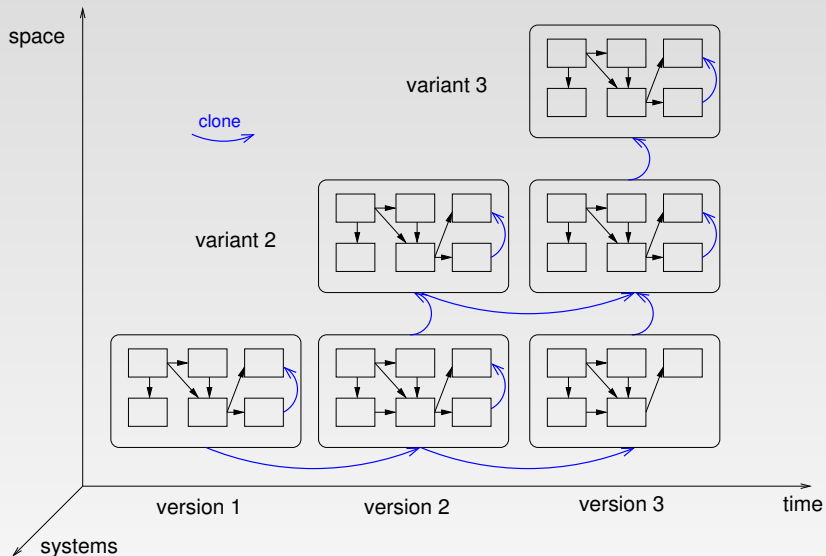


Axivion GmbH [www.axivion.com](http://www.axivion.com)

**Axivion**

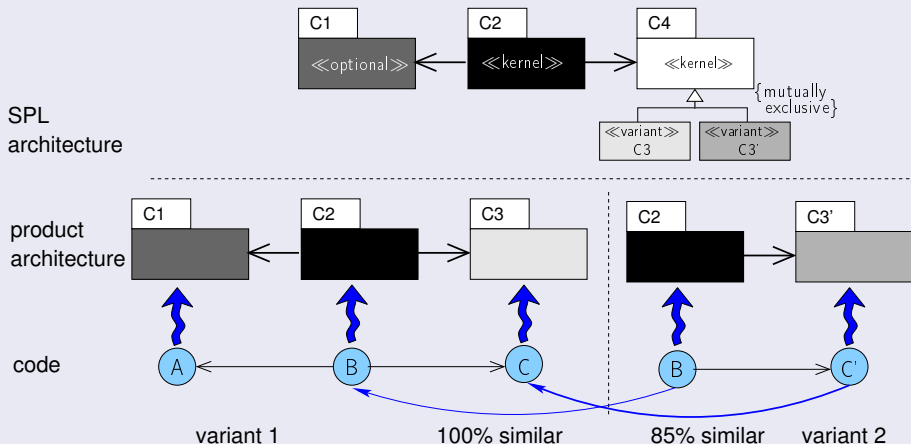
Panel at  
Product LinE Approaches in Software Engineering (PLEASE 2010)  
Cape Town, South Africa  
May, 2nd 2010

# Software Clones in Space and Time



- **Clone Avoidance:**
  - Parameterization
  - Templates/Generics
  - Code Generation
  - Design Patterns
  - ...
- **Required:** Commonalities and variabilities among variants at all levels:
  - source code → **diff & Co., clone detection**
  - architecture → **architecture reconstruction**
  - functionality → **feature location**

## Process



– Koschke u. a. (2009)

- why do (SPL) programmers copy code?
  - why are the existing means being ignored?
    - just ignorance?
    - shortsightedness?
    - incapability to foresee future evolution?
- we failed to deliver suitable means to avoid clones
- clones will be with us
- we need clone-aware program analyses
- we need to improve on consolidating variants into SPLs

[Frenzel u. a. 2007] Frenzel, Pierre ; Koschke, Rainer ; Breu, Andreas P. J. ; Angstmann, Karsten: Extending the Reflection Method for Consolidating Software Variants into Product Lines. In: Working Conference on Reverse Engineering, IEEE Computer Society Press, Oktober 2007, S. 160–169. – Best Paper Award von 87 eingereichten Beitr<E4>gen

[Koschke u. a. 2009] Koschke, Rainer ; Frenzel, Pierre ; Breu, Andreas P. ; Angstmann, Karsten: Extending the reflexion method for consolidating software variants into product lines. 17 (2009), Dezember, Nr. 4, S. 331–366