

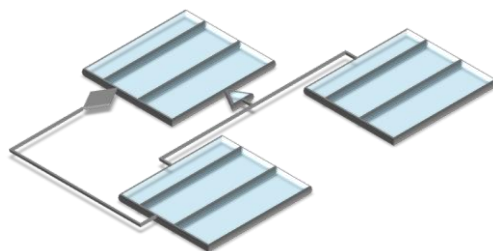
### **Warum ist das interessant?**

Eine Software Produktlinie repräsentiert eine Menge von verwandten Produkten, die auf einer gemeinsamen Basis entwickelt werden, sich aber in ihren Features unterscheiden. Die verfügbaren Features sowie deren valide Kombination werden in der Regel in einem Variabilitätsmodell definiert, durch Code-Artefakte realisiert, und mittels Build-Artefakten für ein spezifisches Produkt zusammengestellt. Um diese Fähigkeiten einer Produktlinie sicherzustellen, existieren spezielle Analysen, die prinzipiell bei jeder Änderung an der Produktlinie erneut ausgeführt werden müssen. Eine Evolutionsanalyse von Open-Source Produktlinien [1,2] hat allerdings gezeigt, dass vergleichsweise wenige Änderungen an den Variabilitätsinformationen vorgenommen werden. Die Frage ist, ob diese Erkenntnis auch für industrielle Software Produktlinien gilt.



### **Was soll ich tun?**

Die Aufgabe besteht darin, sich zunächst in die existierende Infrastruktur zur Commit-Extraktion und -Analyse [3] sowie das kommerzielle Produktlinien-Management-Werkzeug pure-variants [4] einzuarbeiten. Auf dieser Basis sollen dann Konzepte zur Identifikation und Kategorisierung von Änderungen an Code-, Build- und Variabilitätsmodellen in pure-variants definiert werden. Anschließend sollen diese Konzepte realisiert und getestet werden. Dazu muss ggf. zusätzlich eine exemplarische Produktlinie in pure-variants erstellt sowie eine gewisse Evolutionshistorie simuliert werden.



### **Welche Ergebnisse werden erwartet?**

Erwartet wird eine zuverlässige und korrekte Identifikation sowie Kategorisierung von Änderungen an einer Software Produktlinie in pure-variants, so dass eine Evolutionsanalyse auf industriellen Produktlinien anwendbar ist. Weiterhin sollen alle Design-Entscheidungen diskutiert und dokumentiert werden.

### **Was bringt mir das?**

Neben den Kreditpunkten und der Note für den oben genannten Typ dieser Arbeit:

- Einblicke in die Software Produktlinienentwicklung sowie deren Werkzeuge
- Fähigkeiten zur systematischen Identifikation und Realisierung von Konzepten
- Kenntnisse der verwendeten Modellierungs- und Programmiersprachen

[1] Christian Kröher and Klaus Schmid. A Commit-Based Analysis of Software Product Line Evolution: Two Case Studies. Report No. 2/2017, SSE 2/17/E, 2017.

[2] Christian Kröher, Lea Gerling, and Klaus Schmid. Identifying the Intensity of Variability Changes in Software Product Line Evolution. In Proceedings of the 22nd International Systems and Software Product Line Conference, pp. 54-64, ACM, 2018.

[3] Christian Kröher. ComAnI, <https://github.com/CommitAnalysisInfrastructure>, 2018 (last visited 2018-11-30)

[4] pure-systems GmbH. pure-variants, <http://www.pure-systems.com/products/pure-variants-9.html>, 2018 (last visited 2018-11-30).

### **Kontakt**

M.Sc. Christian Kröher  
kroehler@sse.uni-hildesheim.de

### **Aufteilung der Arbeit**

Theorie	Implementierung	Literatur
50%	40%	10%