

State-of-the-Art in Software Product Line Testing and Lessons learned



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Sebastian Oster



Sebastian Oster

oster@es.tu-darmstadt.de
Tel.+49 6151 16 3776

ES Real-Time Systems Lab

Prof. Dr. rer. nat. Andy Schürr

Dept. of Electrical Engineering and Information Technology

Dept. of Computer Science (adjunct Professor)

www.es.tu-darmstadt.de

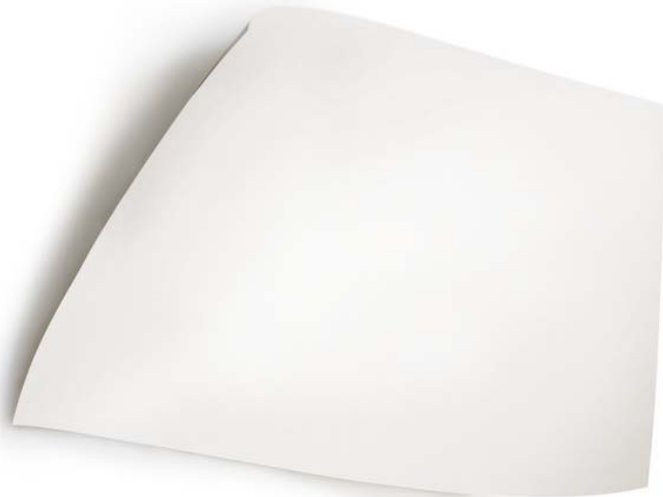
- Ohne Wiederverwendung
- Mit Wiederverwendung
 - Inkrementelles Testen
 - → Wiederverwendung von Testergebnissen
 - Anforderungsbasiertes Testen
 - Z.B. Modellbasiertes Testen
 - → Wiederverwendung von Testartefakten
- Repräsentative Teilmengen
 - Anforderungsbasiert
 - Kombinatorisch



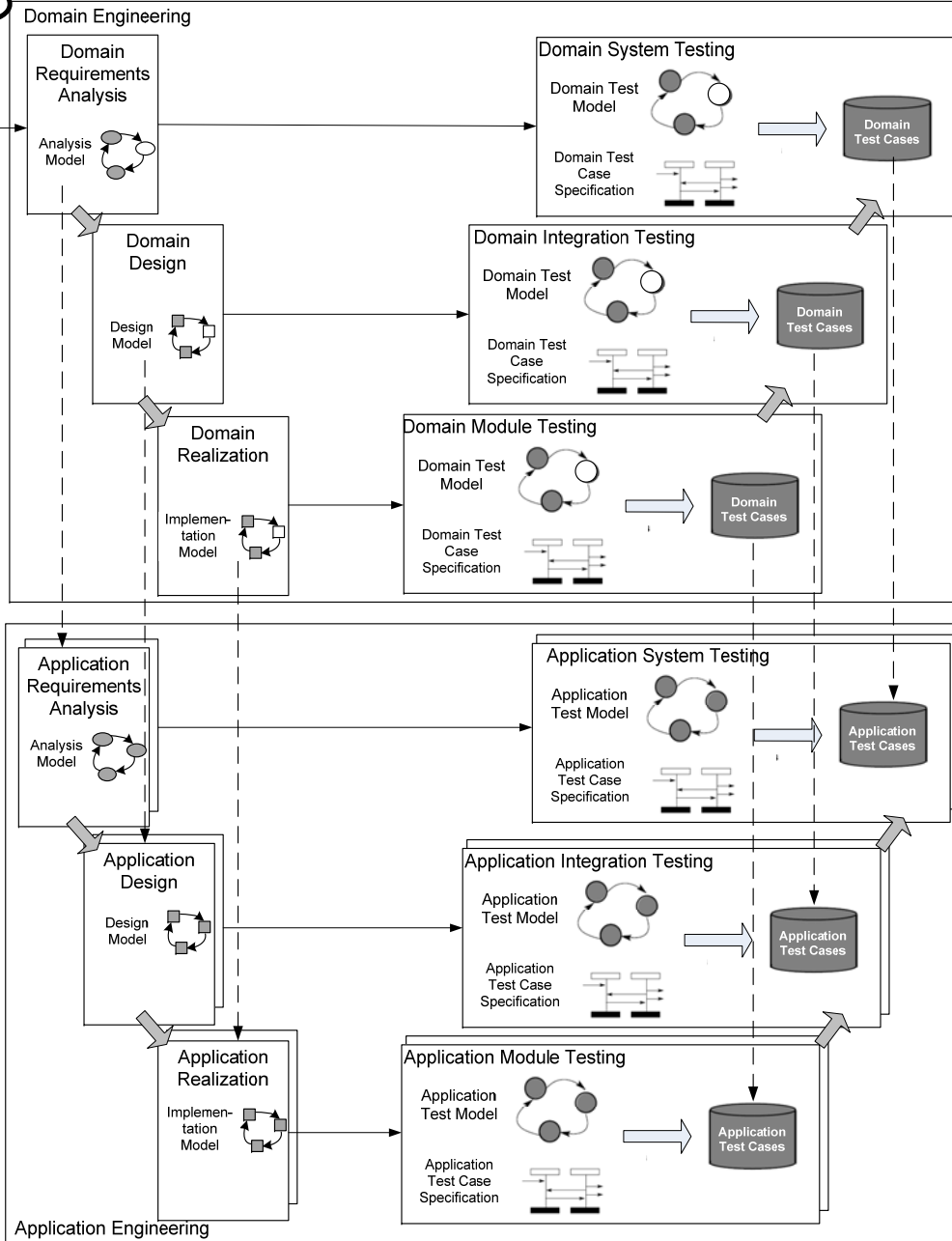
MBT-Ansätze für SPLs



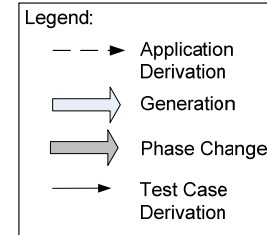
TECHNISCHE
UNIVERSITÄT
DARMSTADT



Elicitation



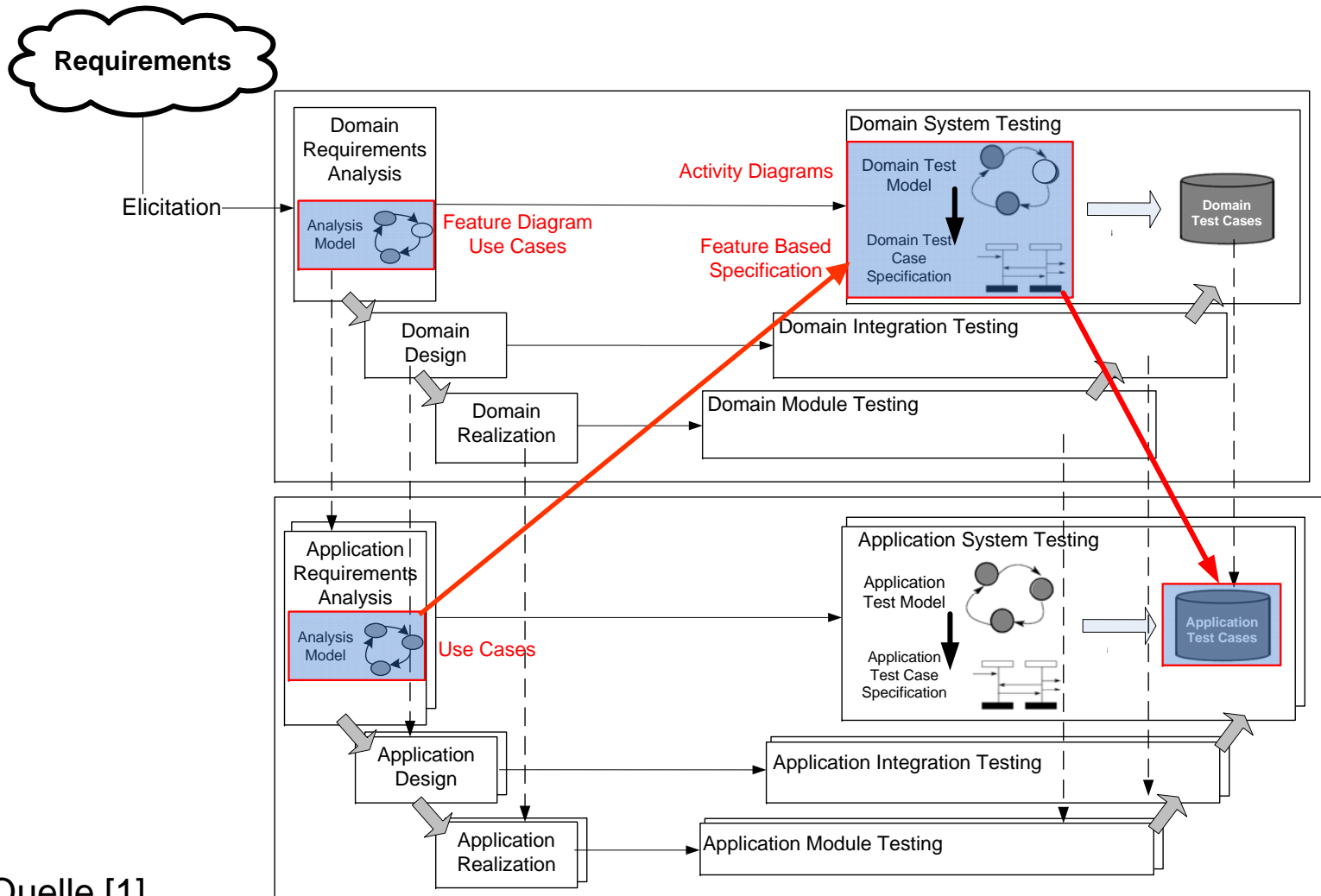
SPL - V-Modell



Siehe Quelle [1]



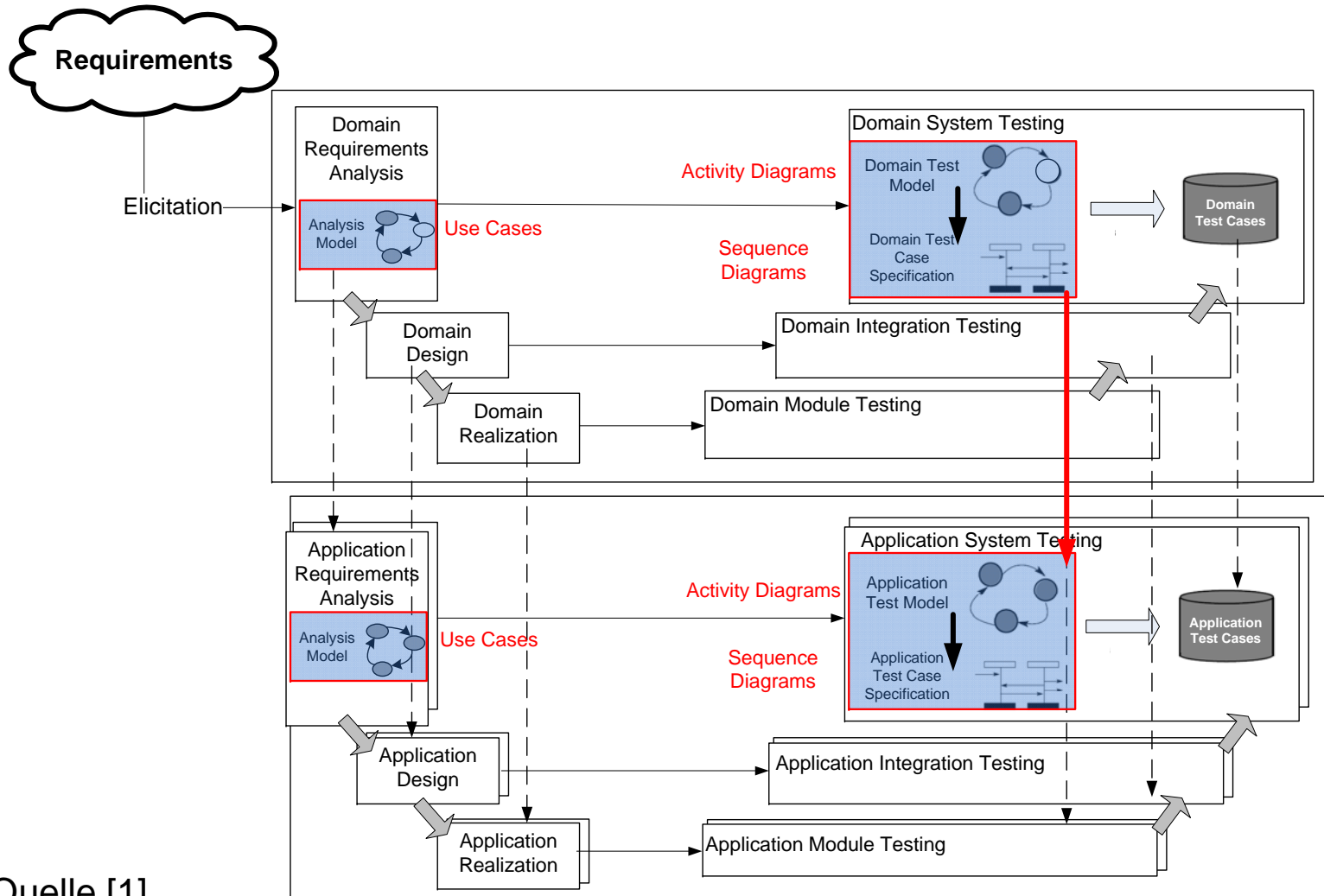
CADeT - Konzept



Siehe Quelle [1]



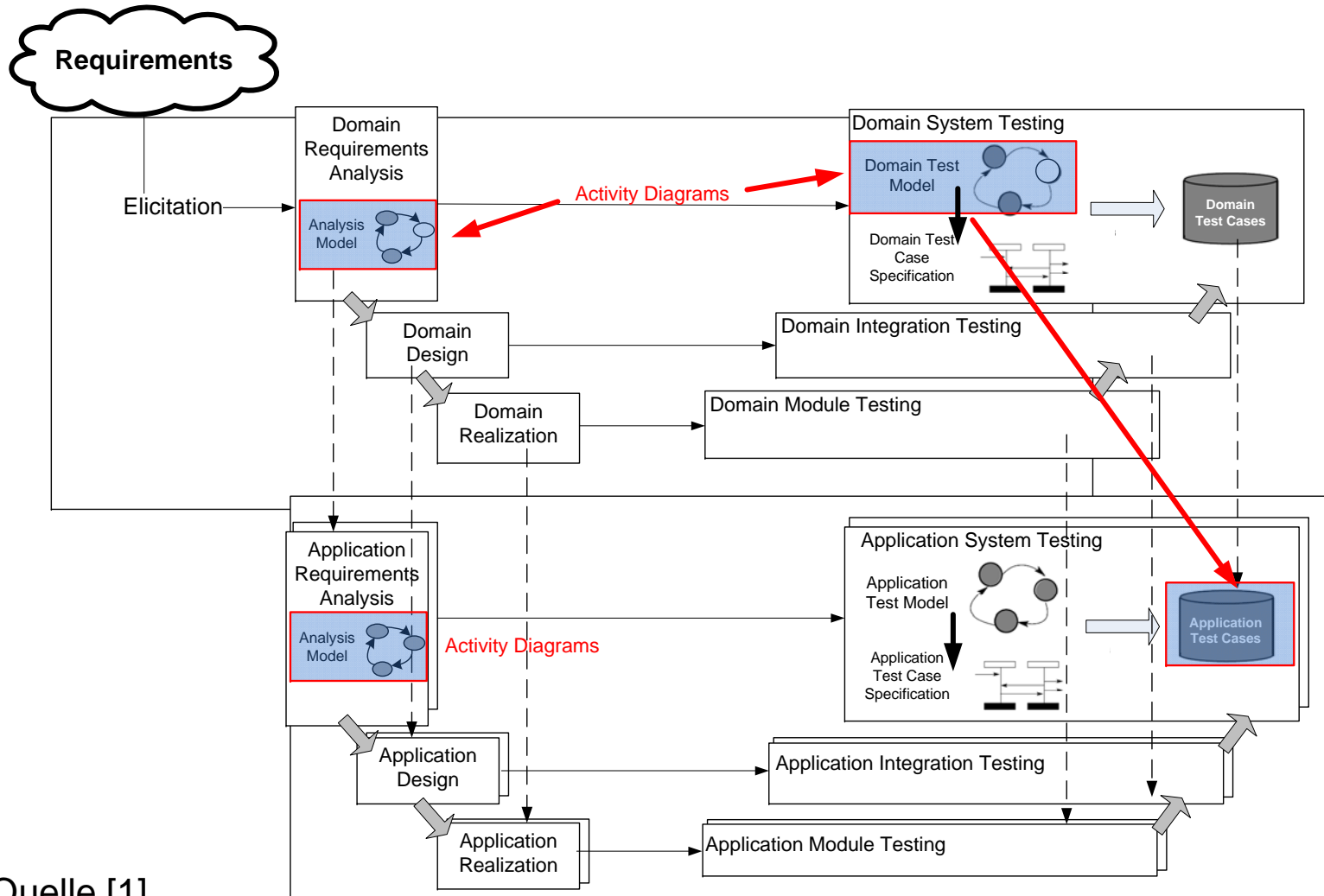
ScenTED - Konzept



Siehe Quelle [1]



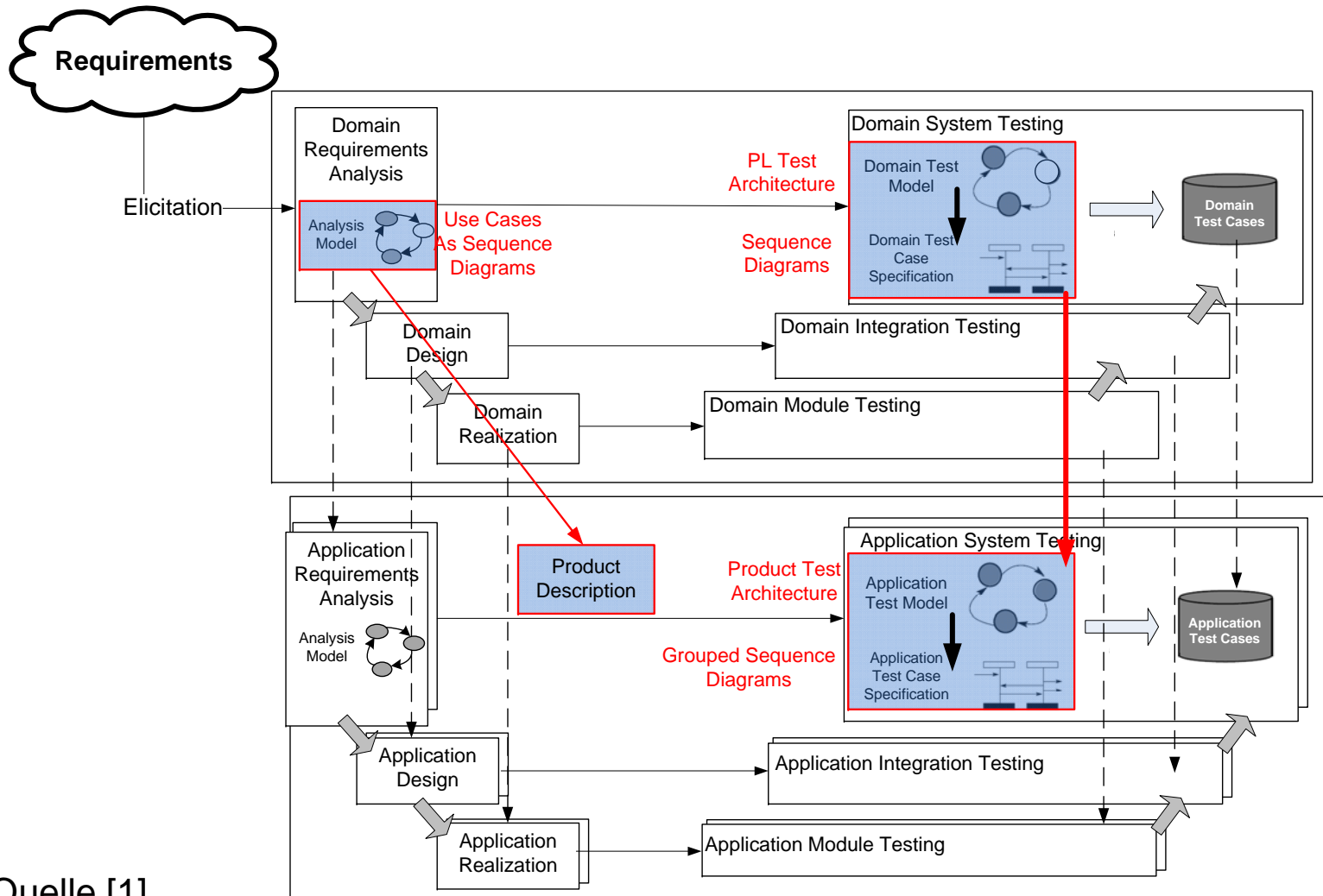
Hartmann et al. - Konzept



Siehe Quelle [1]



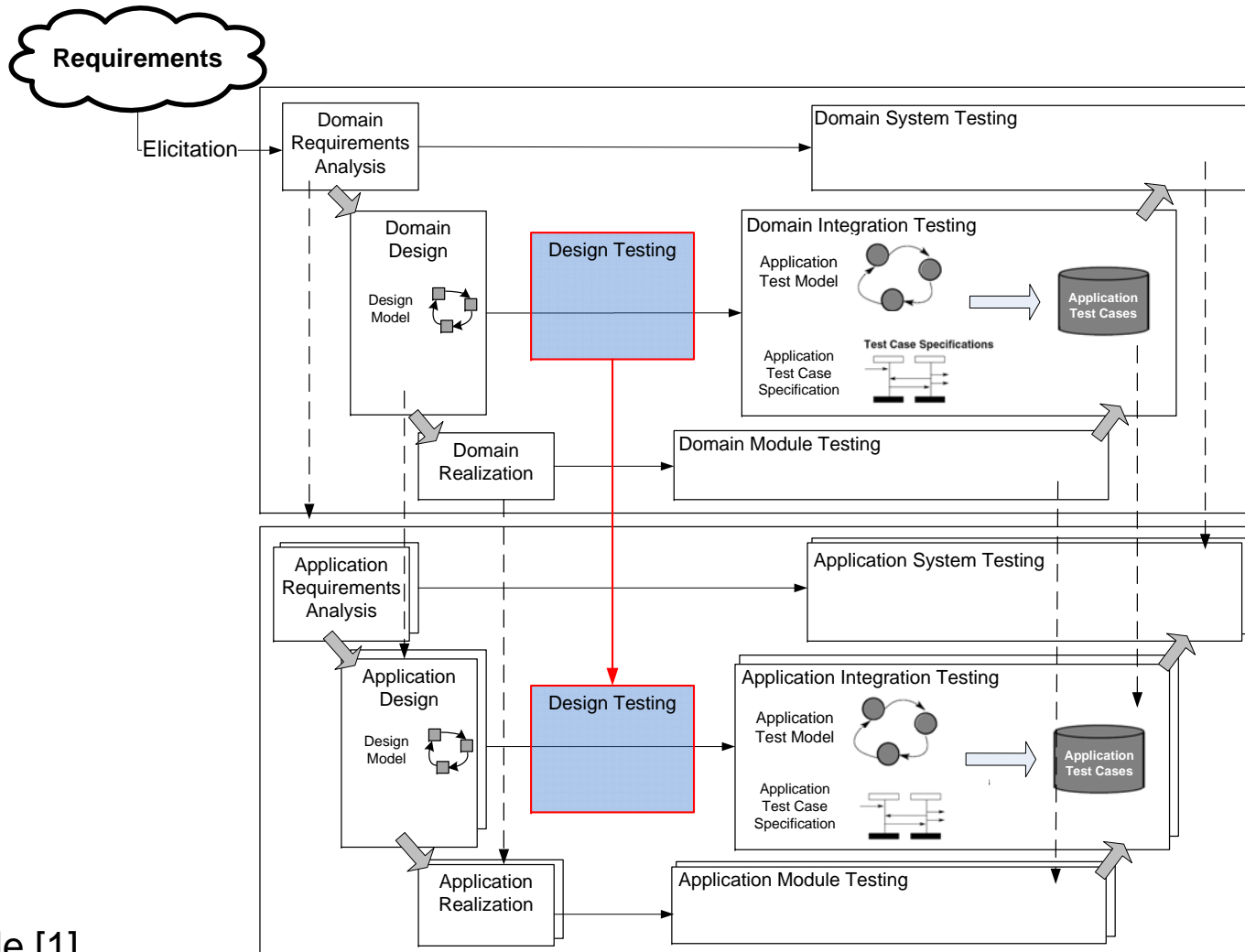
Kang et al. - Konzept



Siehe Quelle [1]



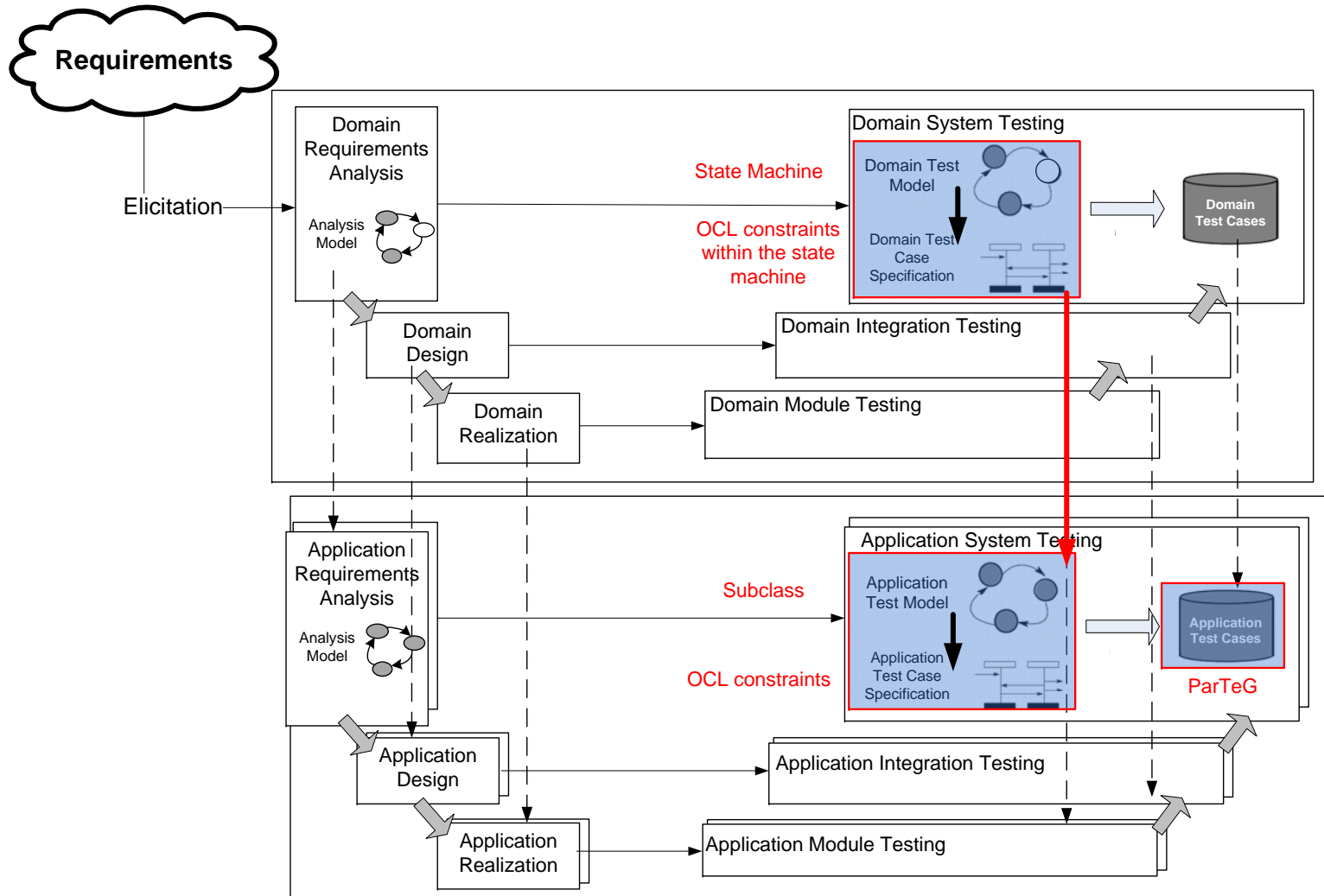
Kishi et al. - Konzept



Siehe Quelle [1]



Weißleder et al. - Konzept



Bewertung der Ansätze

	CADeT -	ScenTED -	UML Based Approach -	Model Checking -	Reusing State Machines -	Framework Approach -
Input - Test Model	Activity Diagram	Activity Diagram	Activity Diagram	State Machine	State Machine	Test Architecture
Input - Dev. Model	Use cases	Use cases	Activity Diagrams	Design Model	not mentioned	Sequence Diagrams
Output - TC - Component	Test Steps pre-, post-condition	Test Steps pre-, and post-cond. input data expected output data	Test Steps	Test Steps input data	Test Steps input data pre- and post-cond.	not mentioned
Output - TC - Automation	(o)	(-)	(-)	(o)	(o)/(+)	(-)
Output - TC - Coverage	not mentioned	branch coverage with variability	not mentioned	not mentioned	boundary based coverage	not mentioned
Test Levels	System Test	System Test	System Test	Design Test	System Test	System Test
Traceability	yes	no	no	no	yes	yes
Dev. Process	Domain & Application Eng.	Application Eng.	Application Eng.	Domain & Application Eng.	Application Eng.	Application Eng.
Dev. Process Integration	(+)	(+)	(-)	(o)	(o)	(+)
Dev. Process - Variability	mandatory optional	mandatory, optional, or, alternative	optional	mandatory, optional, or, alternative	mandatory, optional, or, alternative	mandatory, optional, or, alternative
Application - Instructions	(+)	(o)	(o)	(o)	(o)	(+)



- Systematische Wiederverwendung von Testartefakten
- Testfallgenerierung automatisierbar
- Für Statechart-Ansätze viele Tools verfügbar

ABER:

- Jedes Produkt wird einzeln getestet
- Modelle ggf. umständlich zu erstellen und zu warten → gilt aber für jede Art von Beschreibung des Systemverhaltens

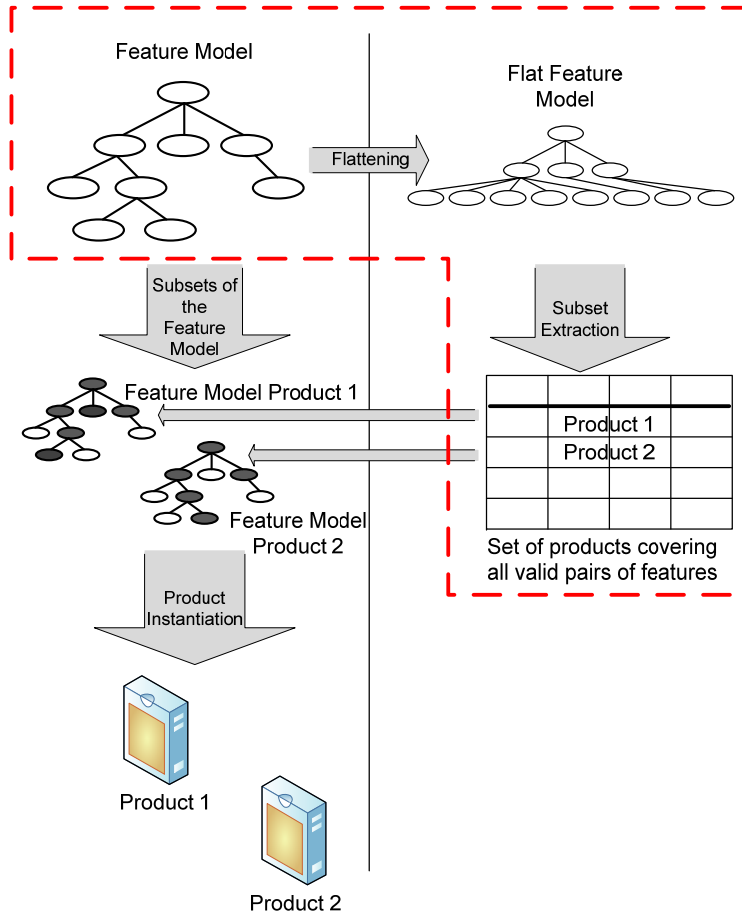
Repräsentative Teilmengen



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Kombinatorisches SPL Testing



→ Generieren einer möglichst kleinen Menge von Konfigurationen, die alle n-wise Feature-Interaktionen abdeckt.

→ Aktuell 2 systematische Ansätze bekannt [2], [3].

Nutzen von kombinatorischen Testen



- Kombinatorisches Testen als „sneak preview“
- Viele Fehler können auf n-wise-Interaktion zurückgeführt werden

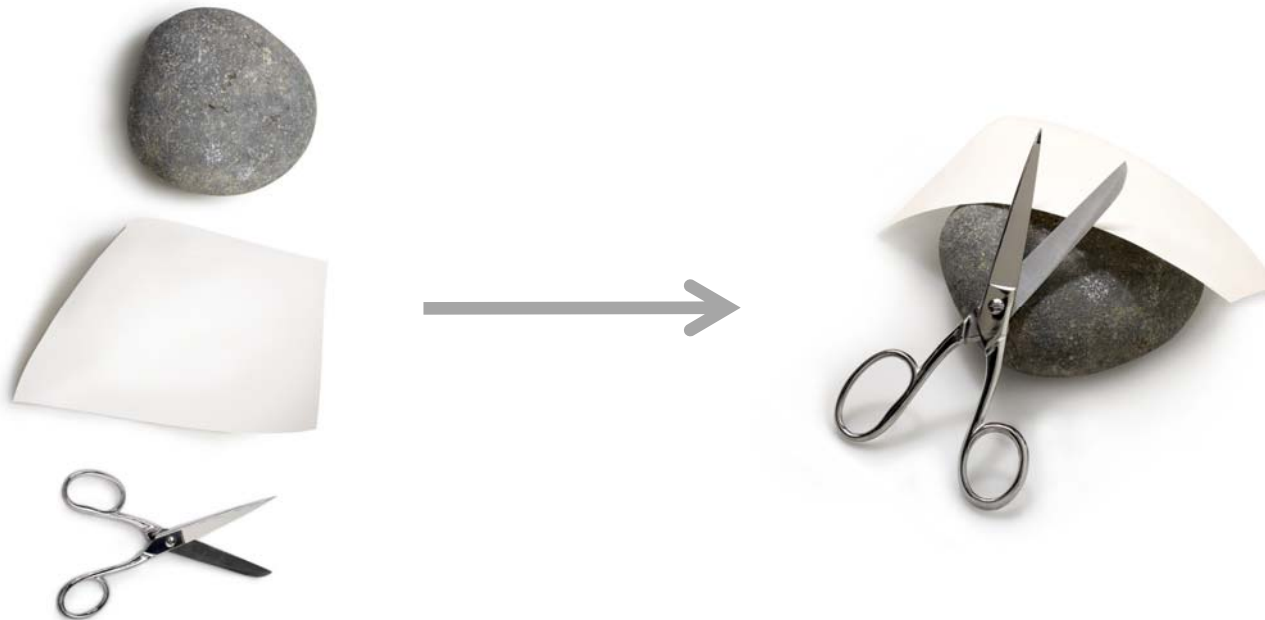
ABER:

- Ansätze berechnen eine Menge an Konfigurationen/Produkten
- Für diese Produkte werden Testfälle benötigt



Integration der Testansätze

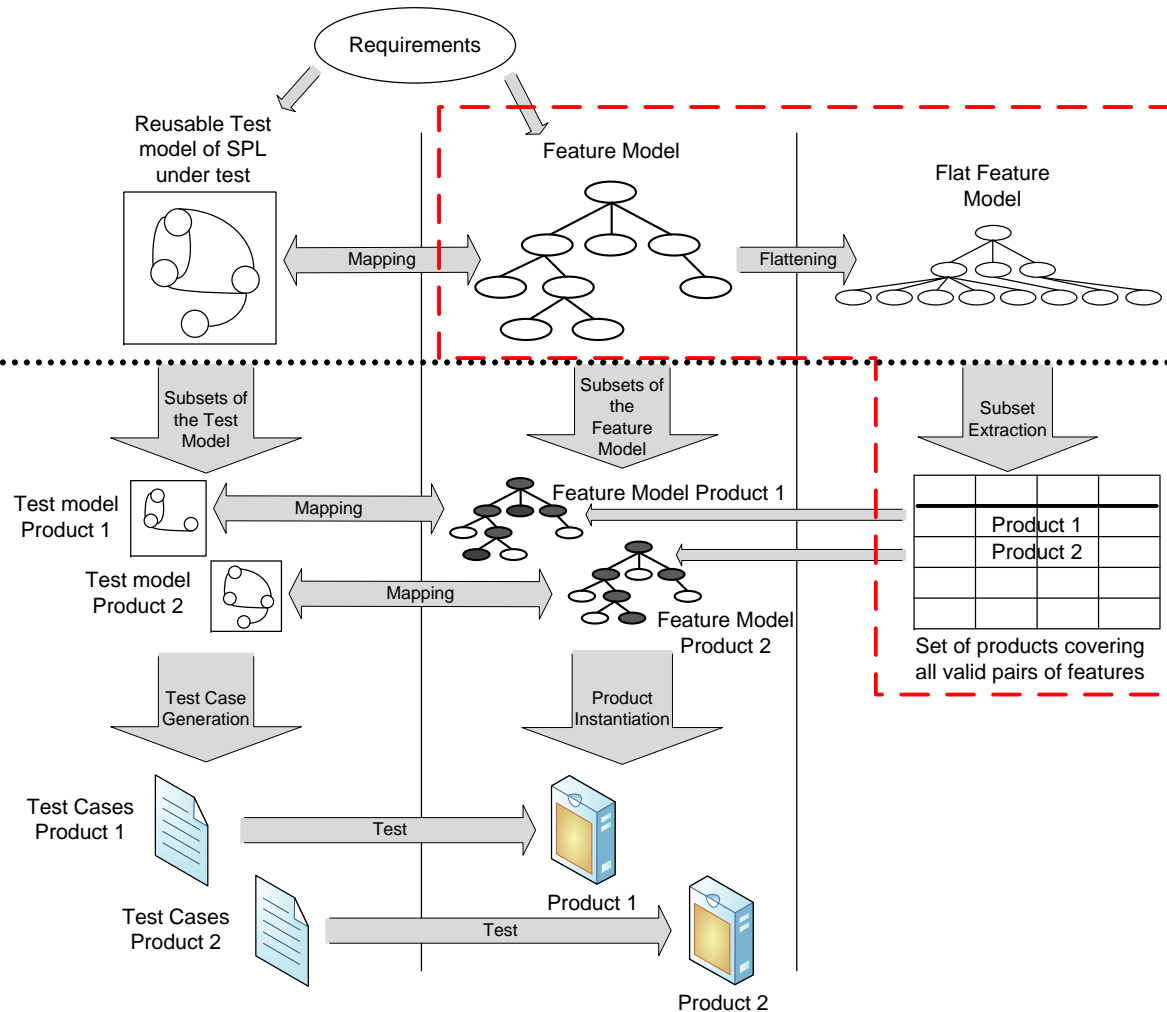
Integration der Testansätze:



MoSo-PoLiTe Konzept

Domain Engineering

Application Engineering



Demo

Bei Interesse an einer Demo bitte email an den Autor.

- [1] **S. Oster, A. Wübbeke, G. Engels, and A. Schürr**, “Model-Based Software Product Lines Testing Survey,” in Modelbased Testing for Embedded Systems, J. Zander, I. Schieferdecker, and P. Mosterman, Eds. CRC Press/Taylor&Francis, 2011, to appear.
- [2] **Oster, S., Ritter, P., Schürr, A.**: Featuremodellbasiertes und kombinatorisches Testen von Software-Produktlinien. In: Proceedings of the SE2010. GI-Edition Lecture Notes in Informatics, Gesellschaft für Informatik (2010)
- [3] **Perrouin, G., Sen, S., Klein, J., Baudry, B., Traon, Y.L.**: Automated and scalable t-wise test case generation strategies for software product lines. In: Third International Conference on Software Testing, Verification and Validation. (2010)