17-708 SOFTWARE PRODUCT LINES: CONCEPTS AND IMPLEMENTATION

ADOPTION AND EVOLUTION

CHRISTIAN KAESTNER
CARNEGIE MELLON UNIVERSITY
INSTITUTE FOR SOFTWARE RESEARCH

LEARNING GOALS

Differentiate and select suitable adoption strategies

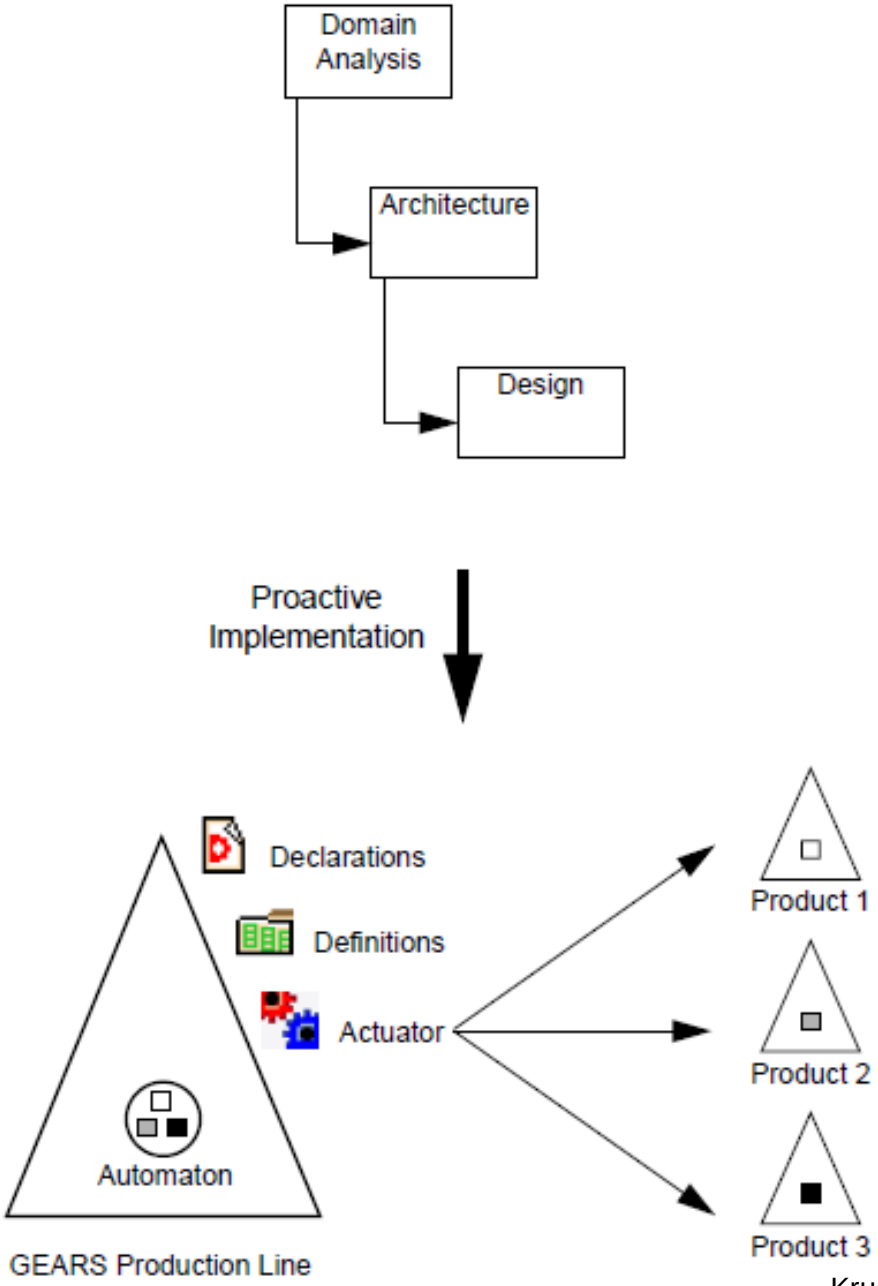
Guild product-line evolution, avoid pitfalls

Understand importance of buy-in and organization aspects

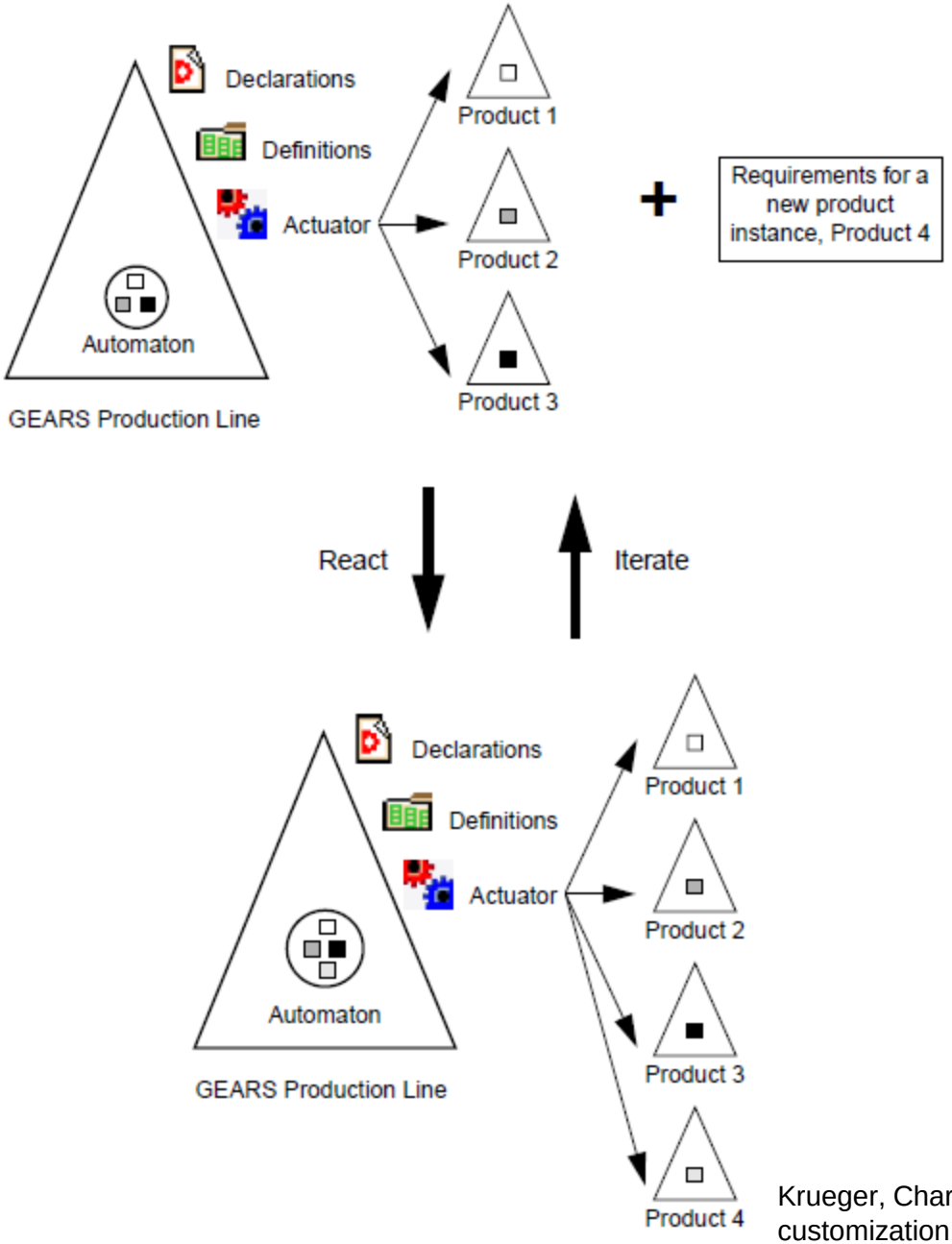
ADOPTION PATHS

Proactive, reactive, extractive

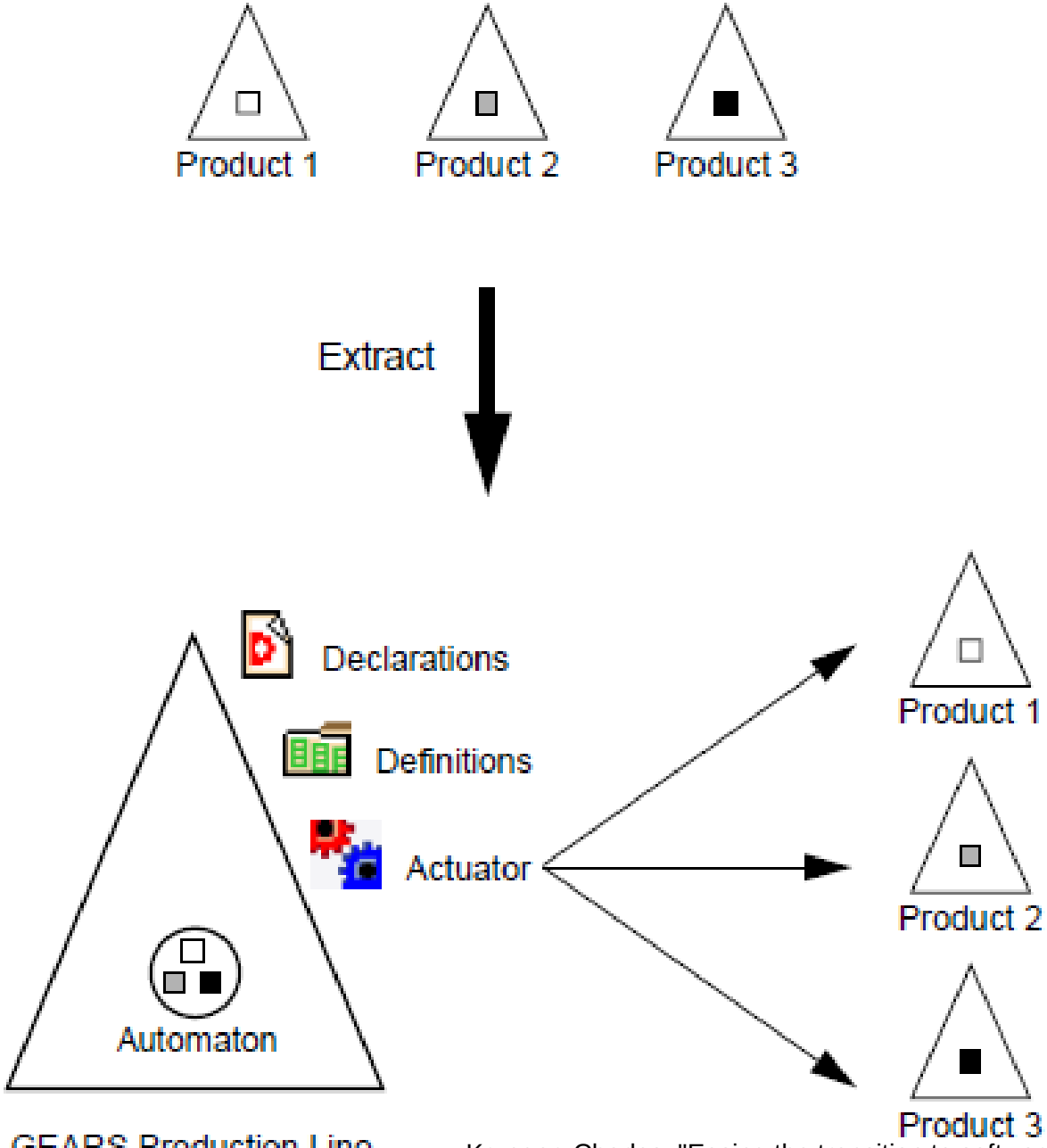
Krueger, Charles. "Easing the transition to software mass customization." In *Software Product-Family Engineering*, pp. 282-293. Springer Berlin Heidelberg, 2002.



Krueger, Charles. "Easing the transition to software mass customization." In *Software Product-Family Engineering*, pp. 282-293. Springer Berlin Heidelberg, 2002.



Krueger, Charles. "Easing the transition to software mass customization." In *Software Product-Family Engineering*, pp. 282-293. Springer Berlin Heidelberg, 2002.



Krueger, Charles. "Easing the transition to software mass customization." In *Software Product-Family Engineering*, pp. 282-293. Springer Berlin Heidelberg, 2002.

TRANSITION STEPS

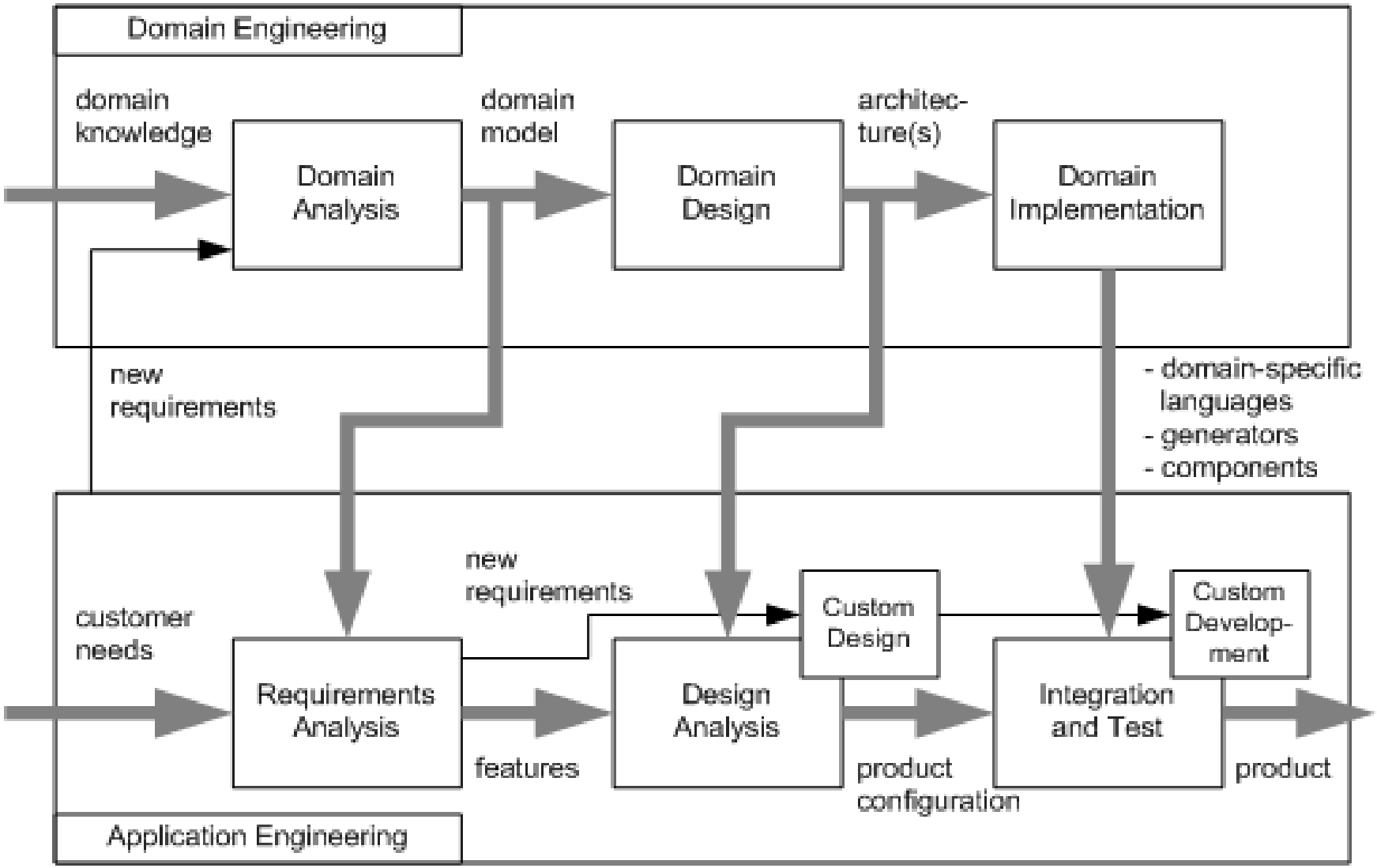
Identify relevant stakeholders

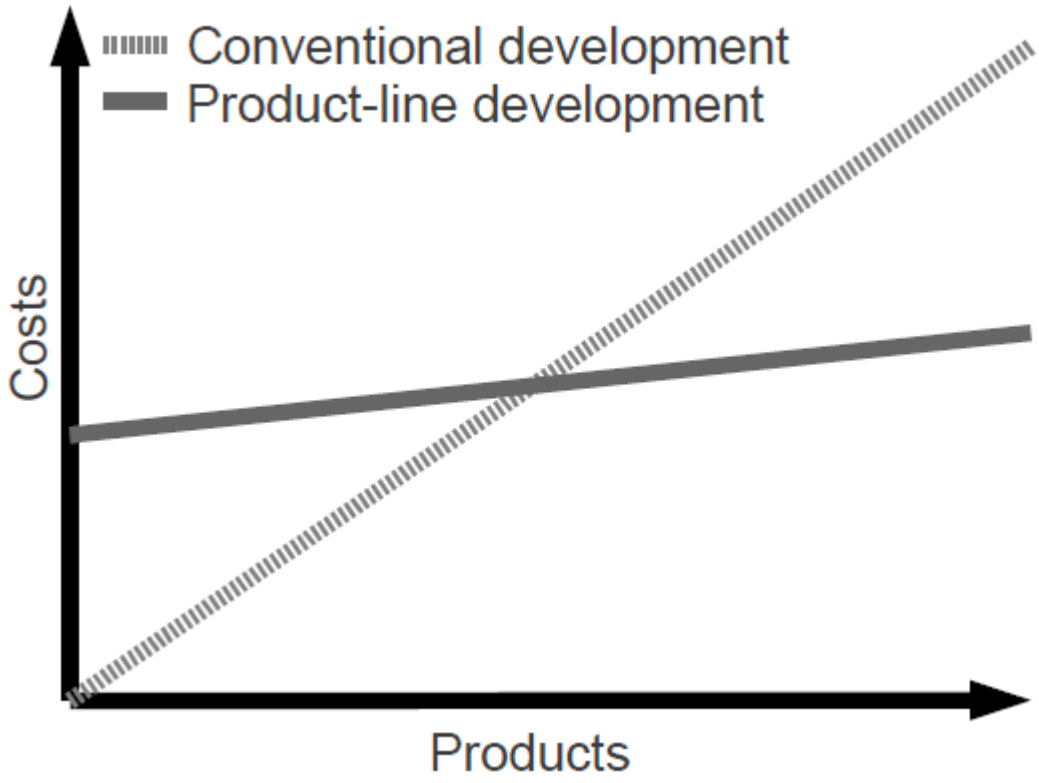
Determine the stakeholders' goals

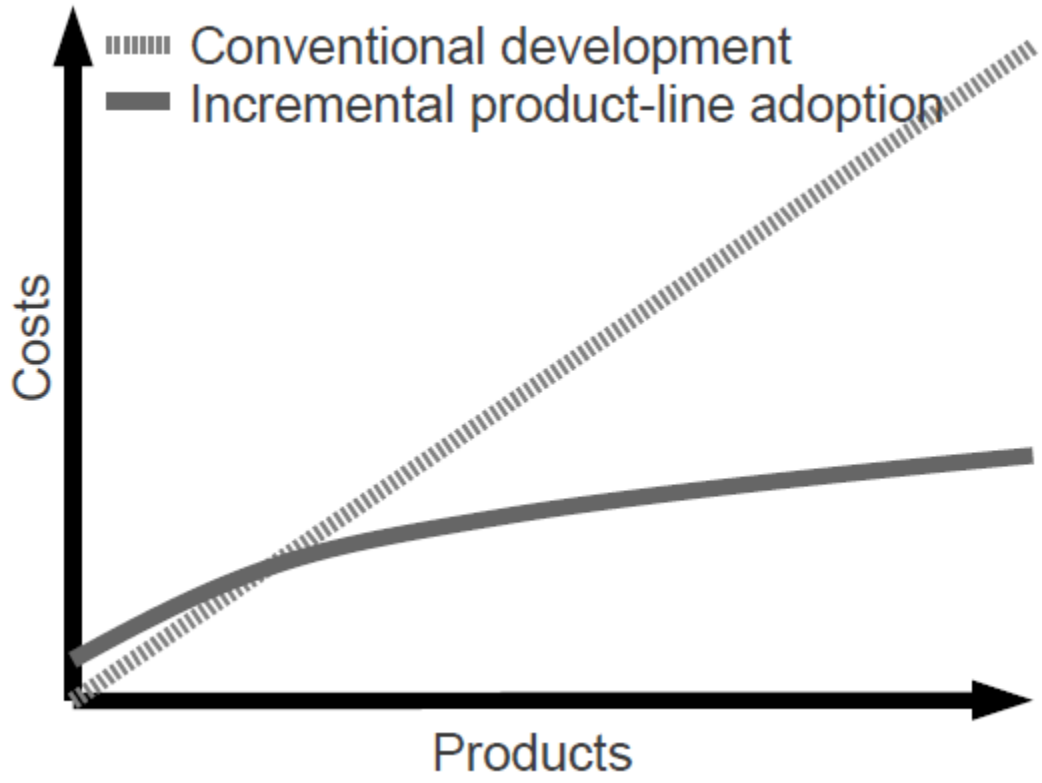
Create business case for all stakeholders

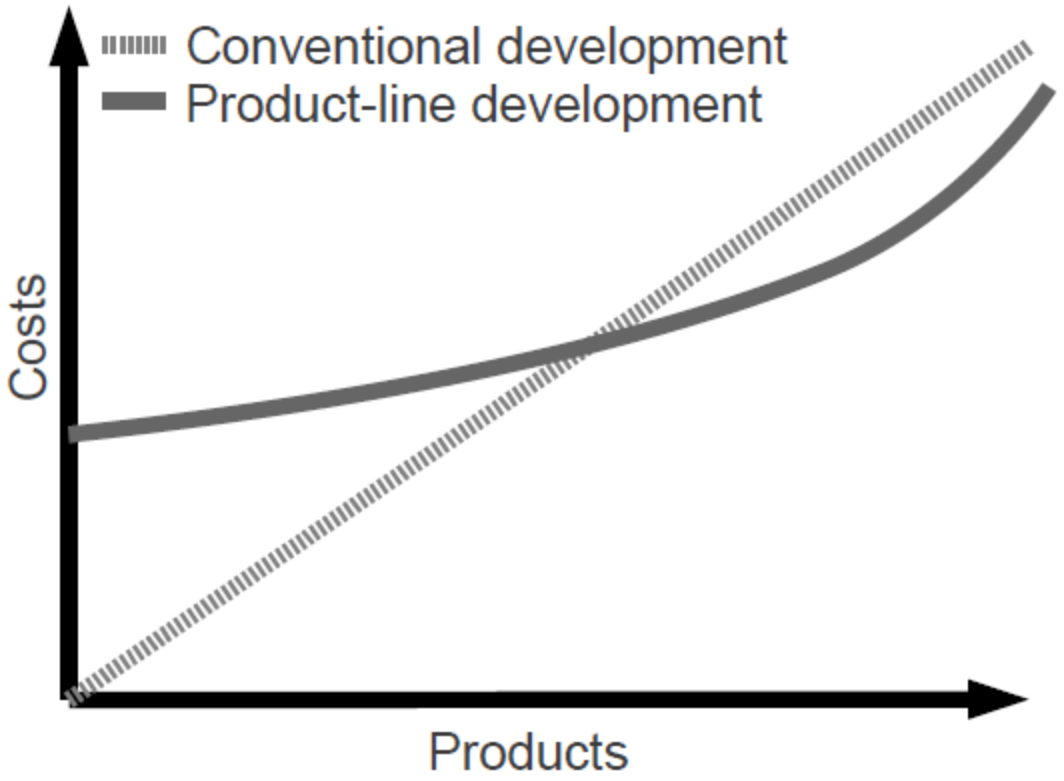
Create an adoption plan

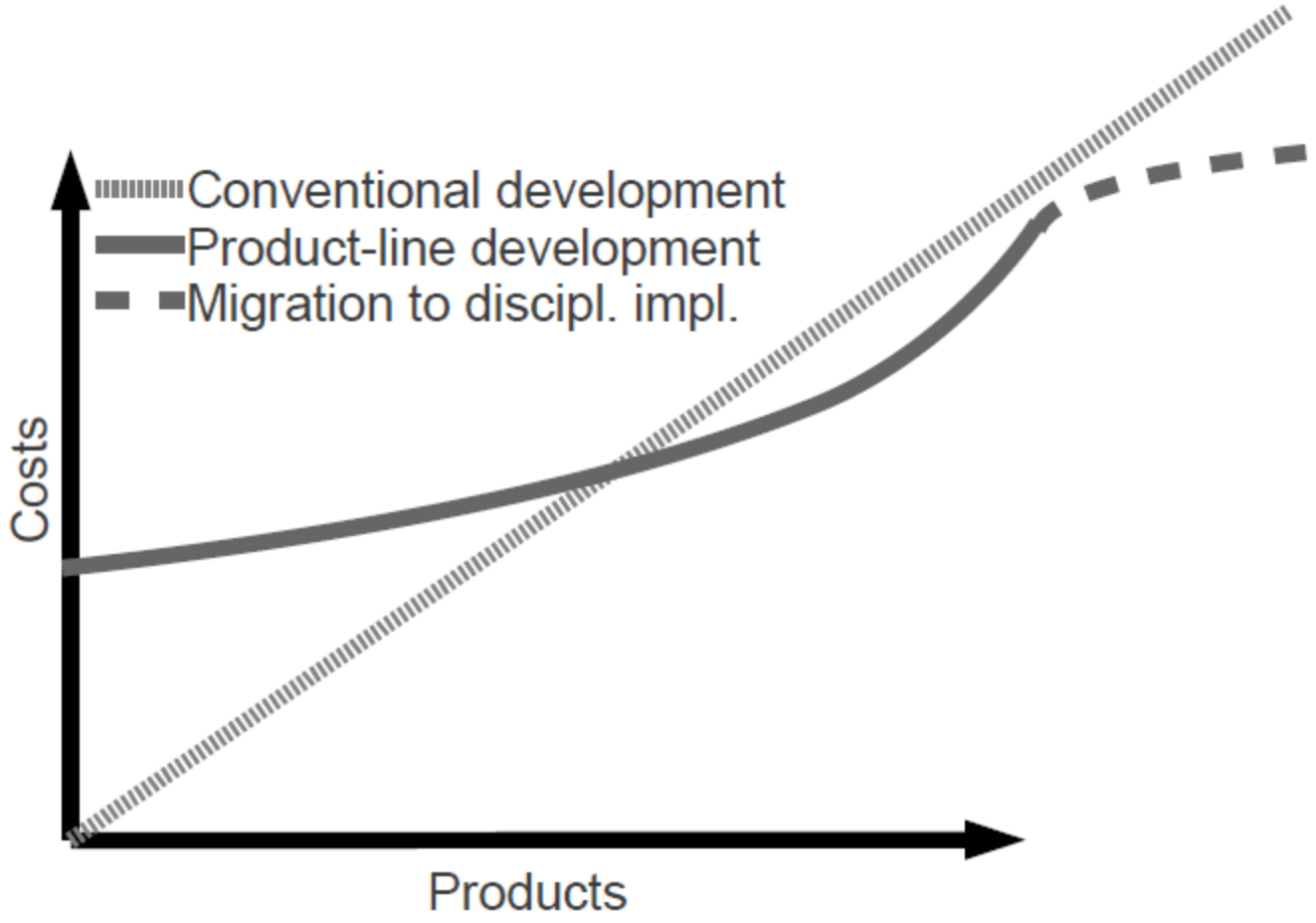
Launching and institutionalizing software product line engineering


Domain Engineering
domain knowledge
Domain Analysis
domain model
Domain Design
architec-ture(s)
Domain Implementation
new requirements
- domain-specific languages
- generators
- components
customer needs
Requirements Analysis
new requirements
Custom Design
Design Analysis
Custom Develop-ment
Integration and Test
features
product configuration
product
Application Engineering


Conventional development
Product-line development
Costs
Products


Conventional development
Incremental product-line adoption
Costs
Products


Conventional development
Product-line development
Costs
Products


Conventional development
Product-line development
Migration to discipl. impl.
Costs
Products

ORGANIZATIONAL ISSUES

Buy in

Technical expertise

Incentives for domain engineering

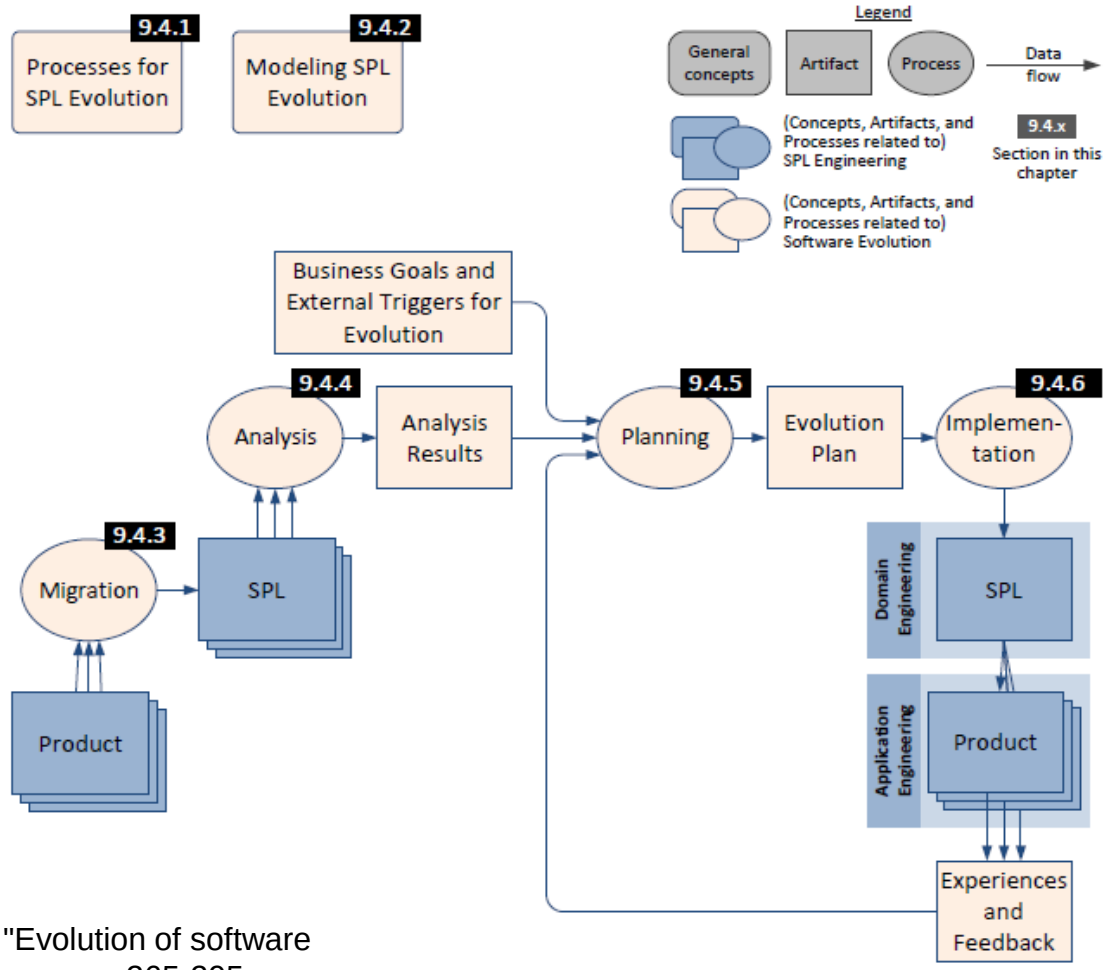
Time for investments into domain artifacts

EVOLUTION IN PRODUCT LINES

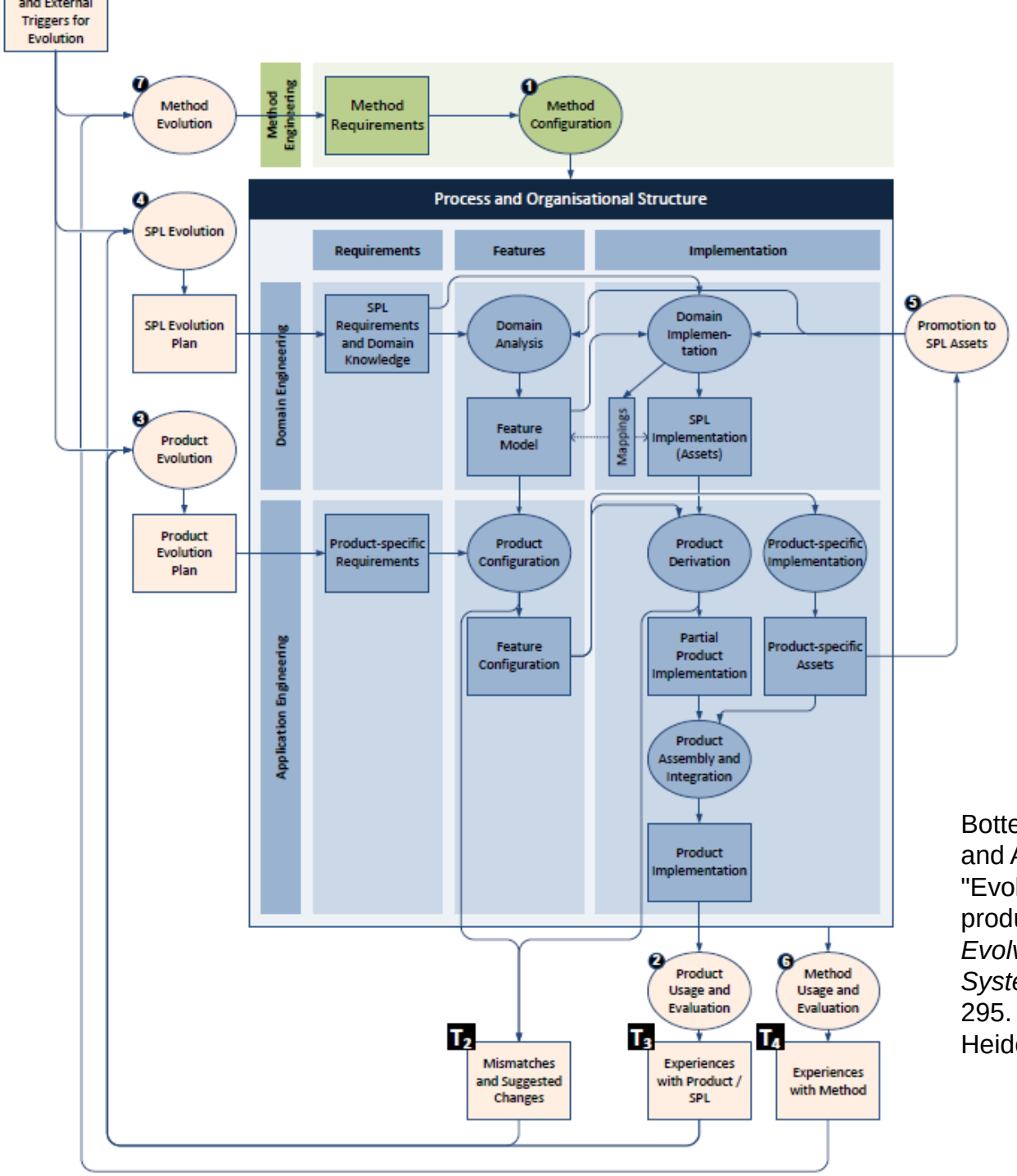
Long life-span

Large size and complexity

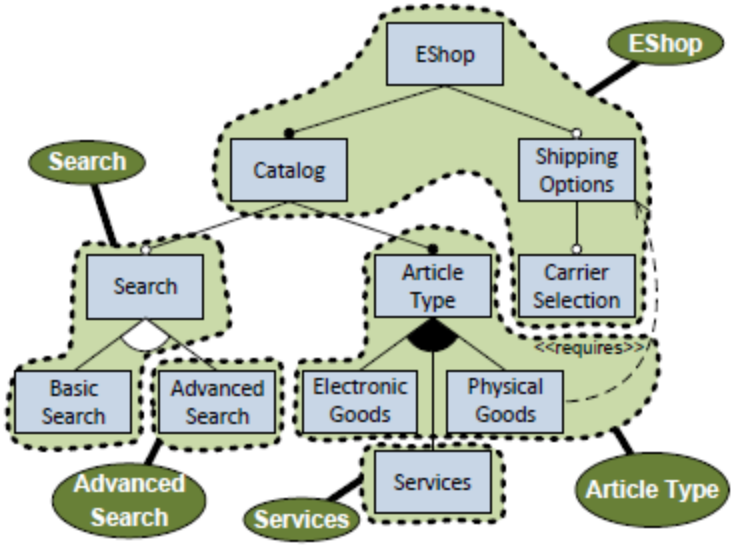
Many interdependencies



Botterweck, Goetz, and Andreas Pleuss. "Evolution of software product lines." In *Evolving Software Systems*, pp. 265-295. Springer Berlin Heidelberg, 2014.



Botterweck, Goetz, and Andreas Pleuss. "Evolution of software product lines." In *Evolving Software Systems*, pp. 265-295. Springer Berlin Heidelberg, 2014.



(a) Clustering into fragments

Fragment \ Time	2012	Current 2013	2014	2015
EShop				
Search				
Advanced Search				
Article Type				
Services				
<ShippingOptions optional>				

(b) Evolution plan

Botterweck, Goetz, and Andreas Pleuss. "Evolution of software product lines." In *Evolving Software Systems*, pp. 265-295. Springer Berlin Heidelberg, 2014.

REFACTORING

Variability-preserving vs variability enhancing refactorings

- Splitting, merging of features
- Refinement

Stepwise migration?

- Introduction of abstractions

Coevolution of feature model and implementation artifacts

COSTS OF PRODUCT LINES

Reaction speed? Flexibility?

Flexibility within domain, but slow movement toward new domains?

-> Nokia history

Product line is always outdated

"A product line is a fantastic approach to create recently outdated products fast"

"Automation can make creation of the obsolete products very efficient"

MODULARITY AND CHANGE

Backward compatibility

Open vs closed world

FURTHER READNG

Pohl, Klaus, Günter Böckle, and Frank J. van der Linden. *Software product line engineering: foundations, principles and techniques*. Springer Science & Business Media, 2005. Part V

Botterweck, Goetz, and Andreas Pleuss. "Evolution of software product lines." In *Evolving Software Systems*, pp. 265-295. Springer Berlin Heidelberg, 2014.