

Software product lines

Paulo Borba

Informatics Center

Federal University of Pernambuco

Practices and processes for software product lines

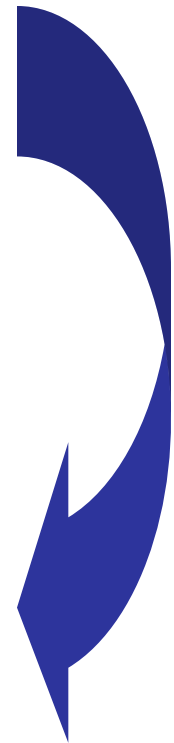
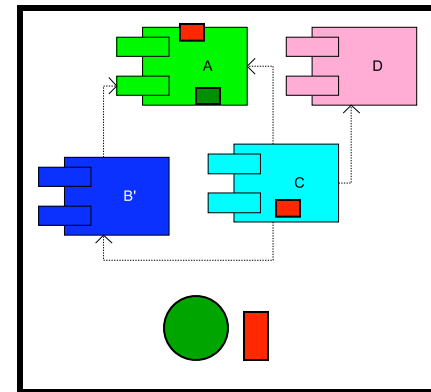
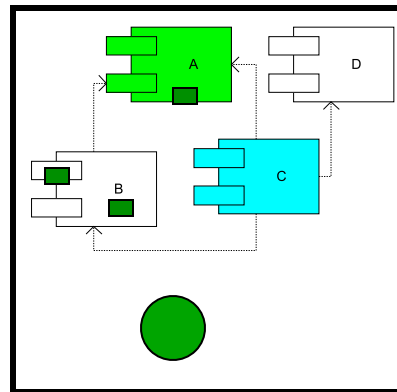
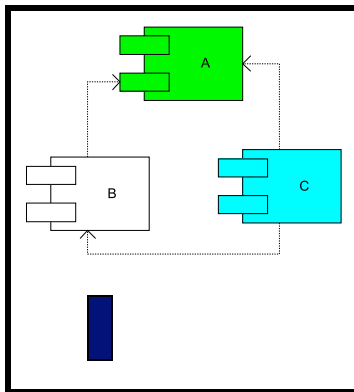
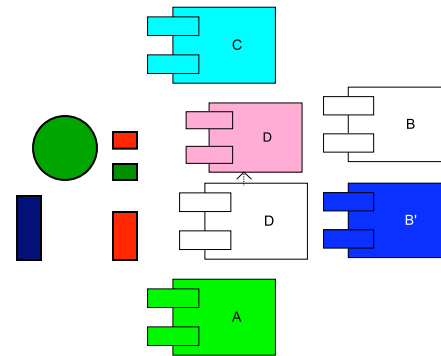
Paulo Borba

Informatics Center

Federal University of Pernambuco

Product lines, how to develop one?

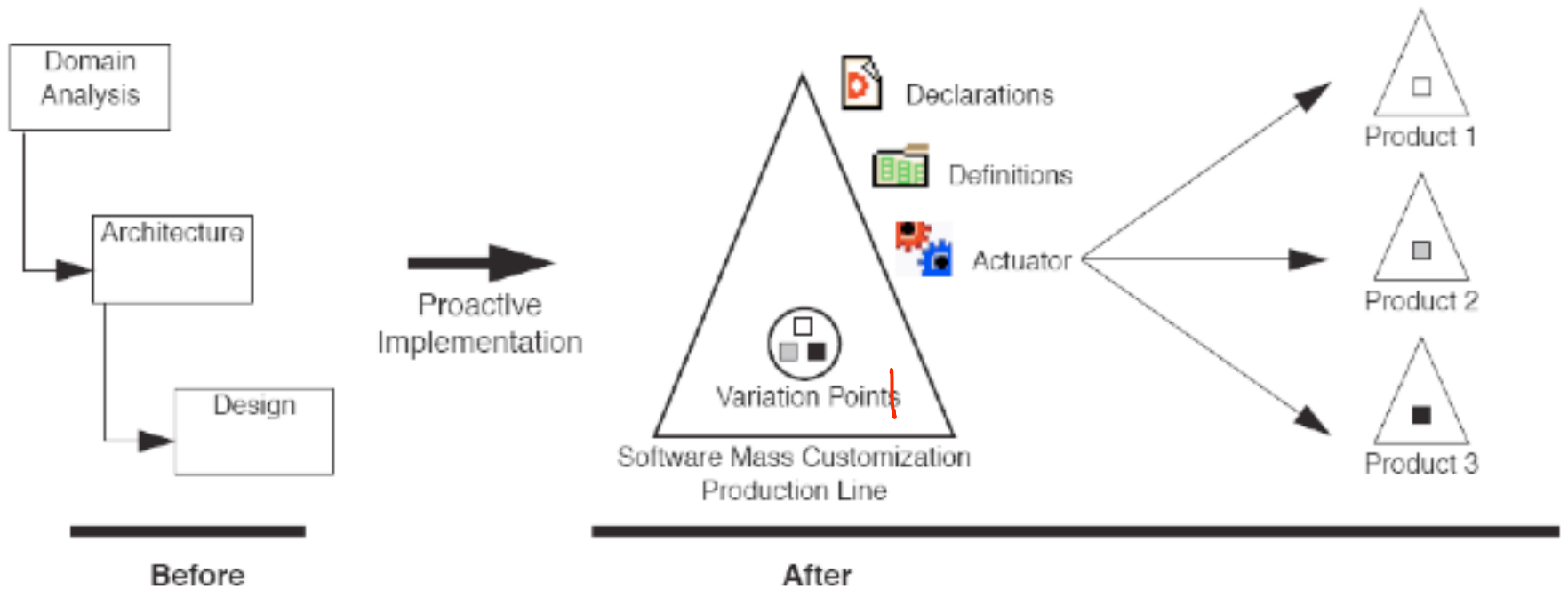
Strategic reuse of common artifacts



Different adoption strategies

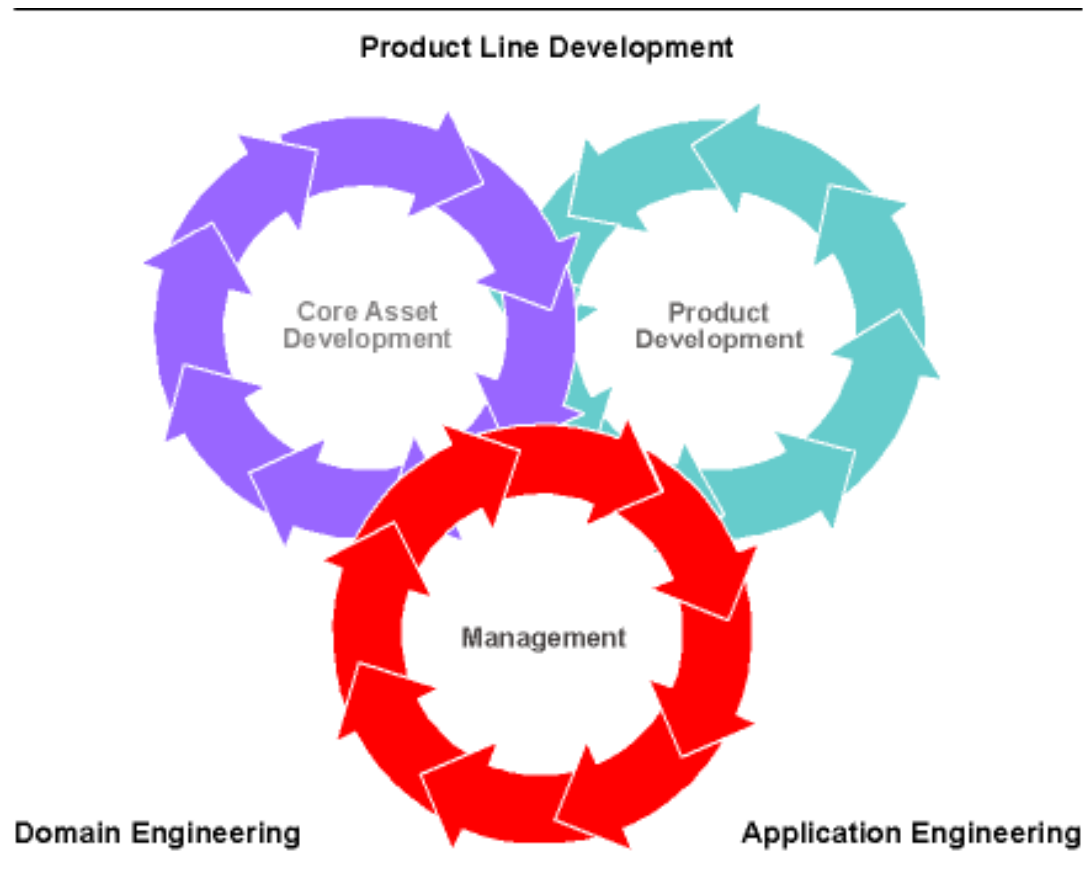
- Proactive
- Extractive
- Reactive

Proactive adoption strategy



Source: Krueger, PFE'01

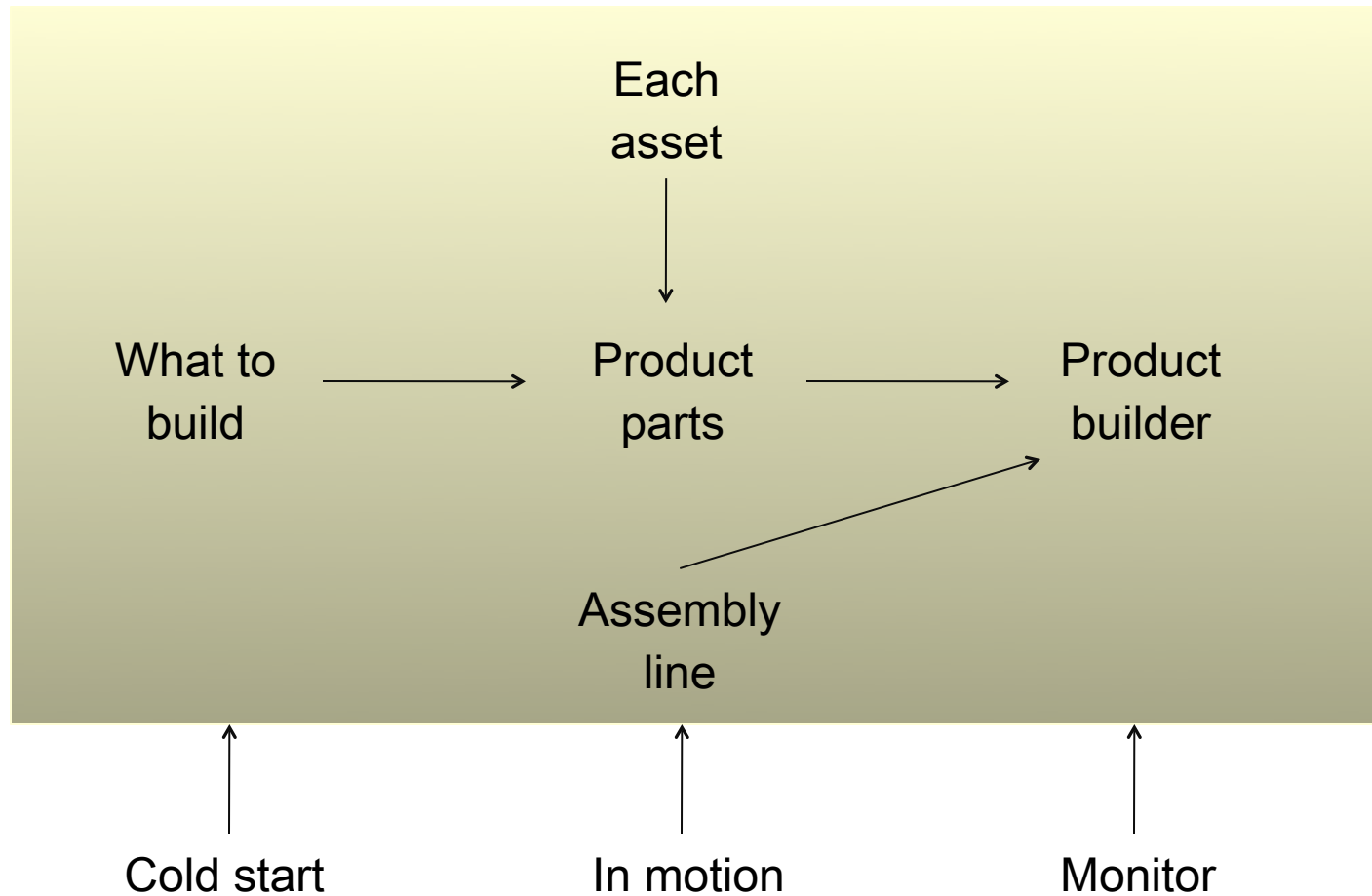
Domain and application engineering



Source: SEI

Factory pattern

(Clements and Northorp 2002)



Product management

- Market and product strategy
- Portfolio management, scoping
- Product definition
- Product support
- Market introduction
- Market observation
- Product controlling

Domain requirements engineering

- Traditional activities
 - elicitation
 - documentation
 - validation
- Commonality and variability analysis
 - traditional artifacts plus existing applications
 - application requirements matrix
 - checklist based analysis

Variability modelling

- Features
 - dependences among sibling features
 - dependences among other features
- Variations
 - architects involved
- Variation points
- Relation between features and artifacts

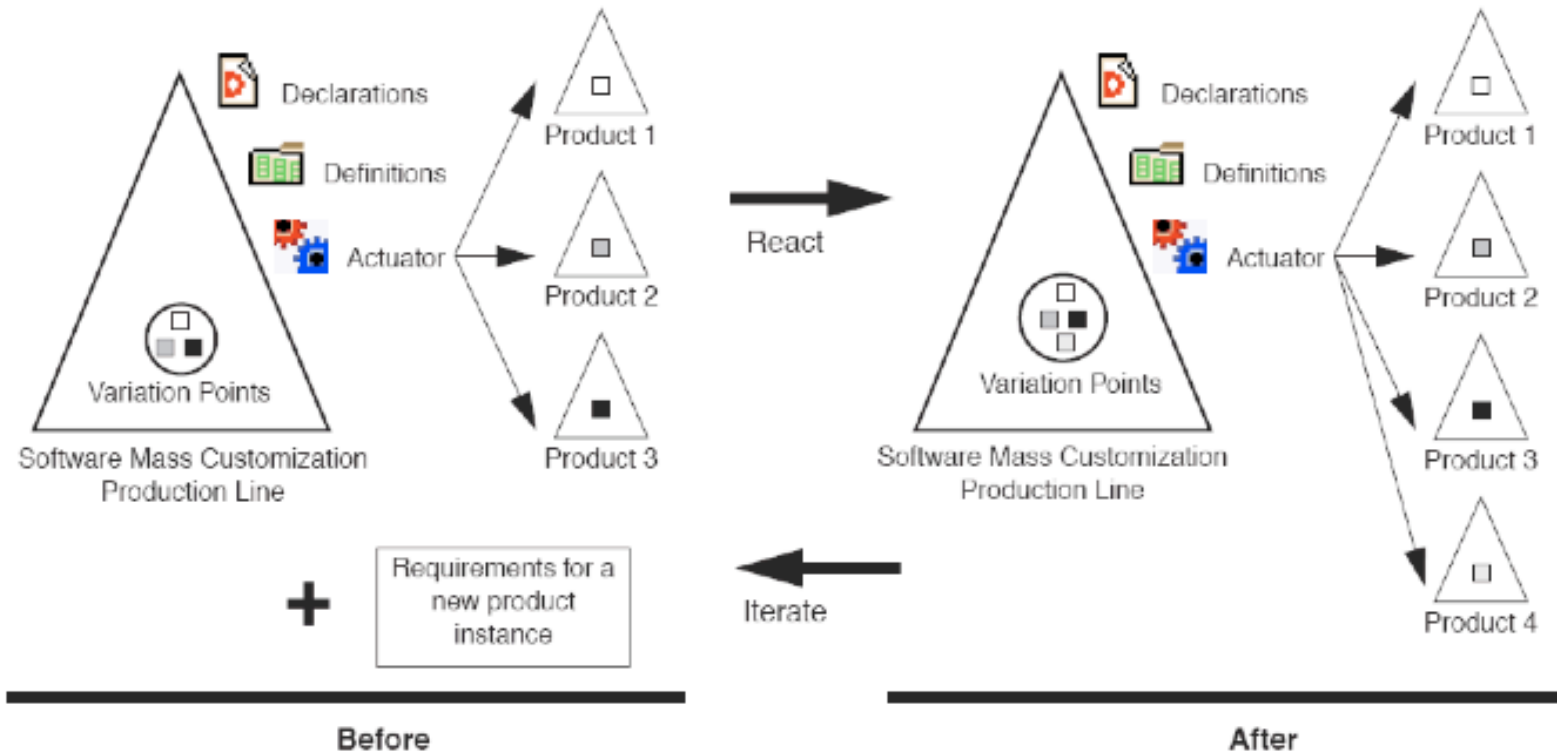
For planned applications,
there is no application
requirements engineering...

feature model instantiation
=
instance model creation

Application requirements engineering

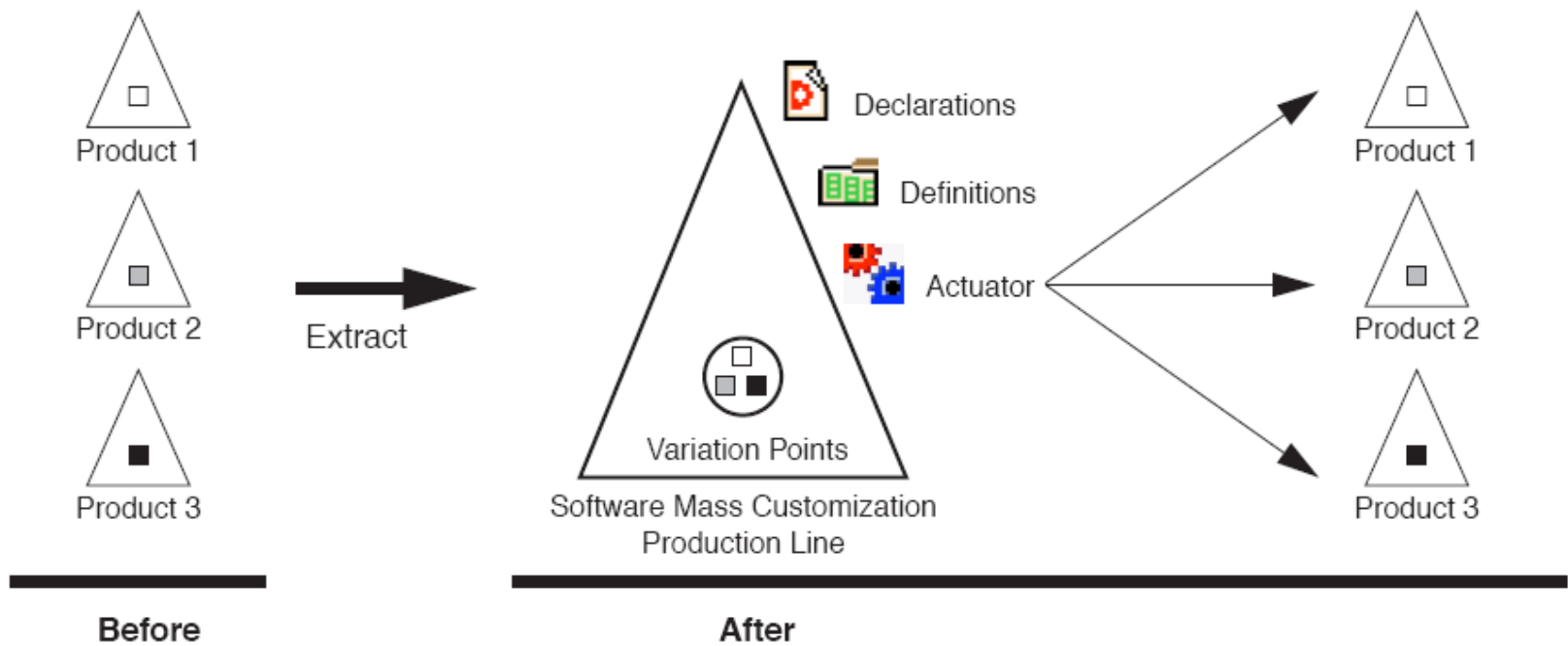
- Traditional activities
- Communication of the product line variability
- Analysis of requirements deltas
- Impact on domain artifacts
 - changes (refactorings, inclusive to FM)
 - additions
- Instance model creation

Reactive adoption strategy



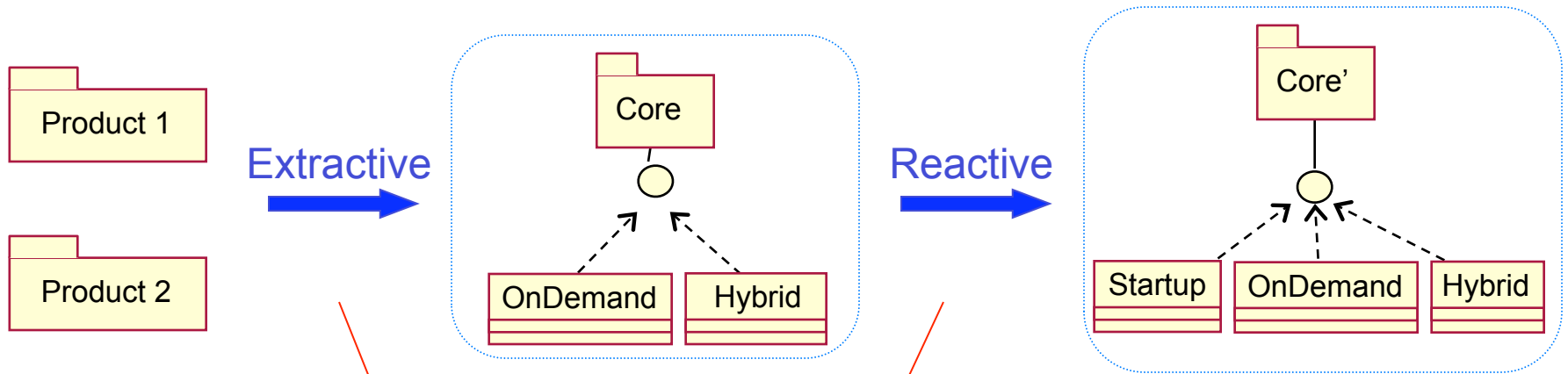
Source: Krueger, PFE'01

Extractive adoption strategy



Source: Krueger, PFE'01

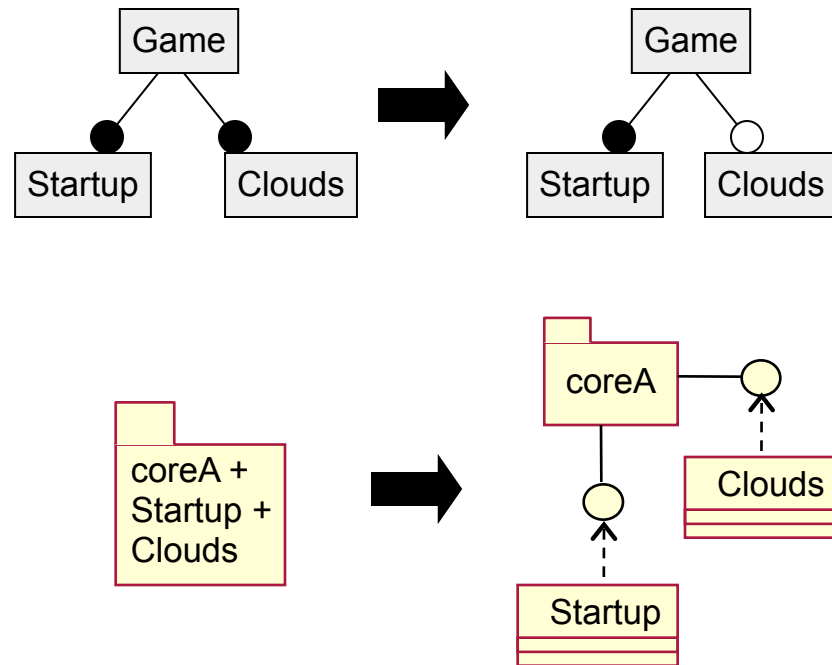
Product line derivation and evolution benefit from refactoring



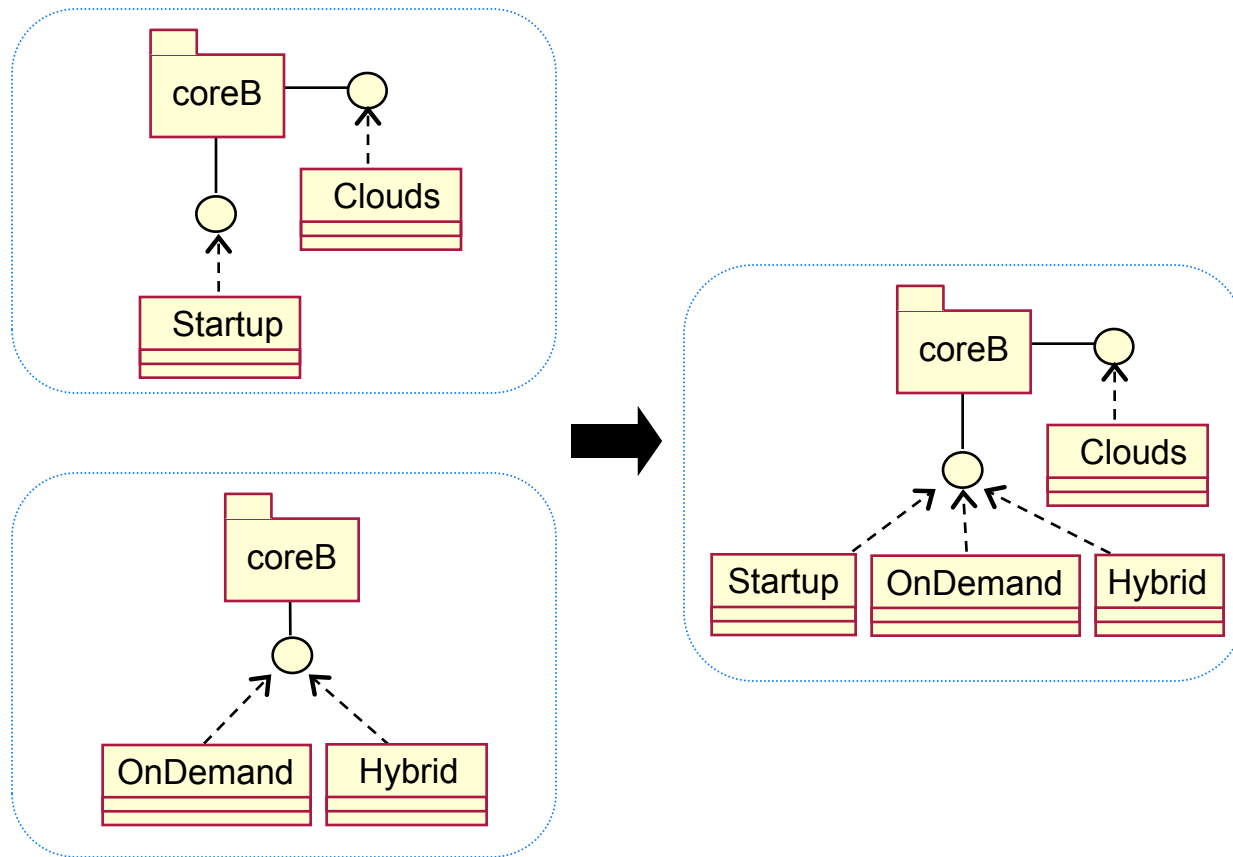
Guidelines and safety

But PL refactoring should go beyond code...

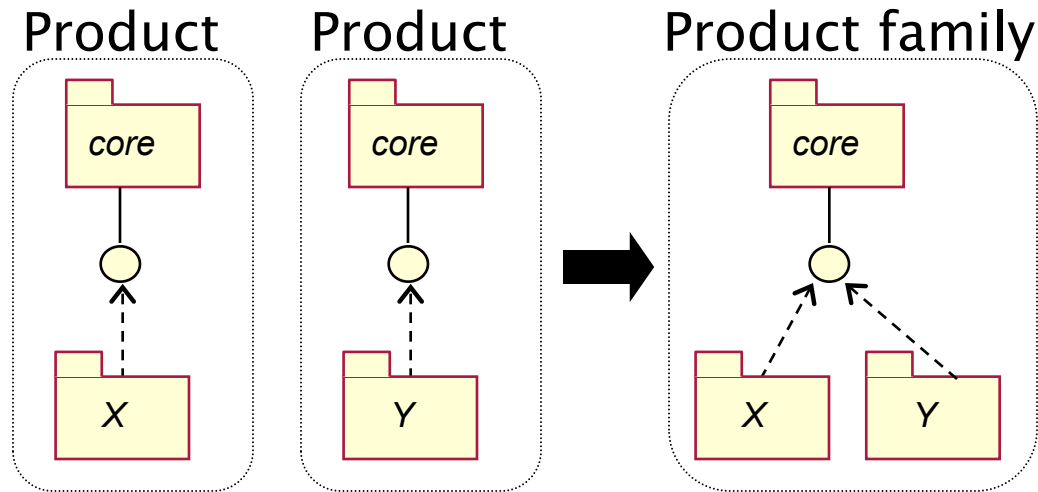
Rain of fire



and deal with populations and families too!



Refactoring populations and families: code



$$p1 \ p2 \sqsubseteq pL =$$

$$\exists p, q \in pL \bullet p1 \sqsubseteq p \wedge p2 \sqsubseteq q$$

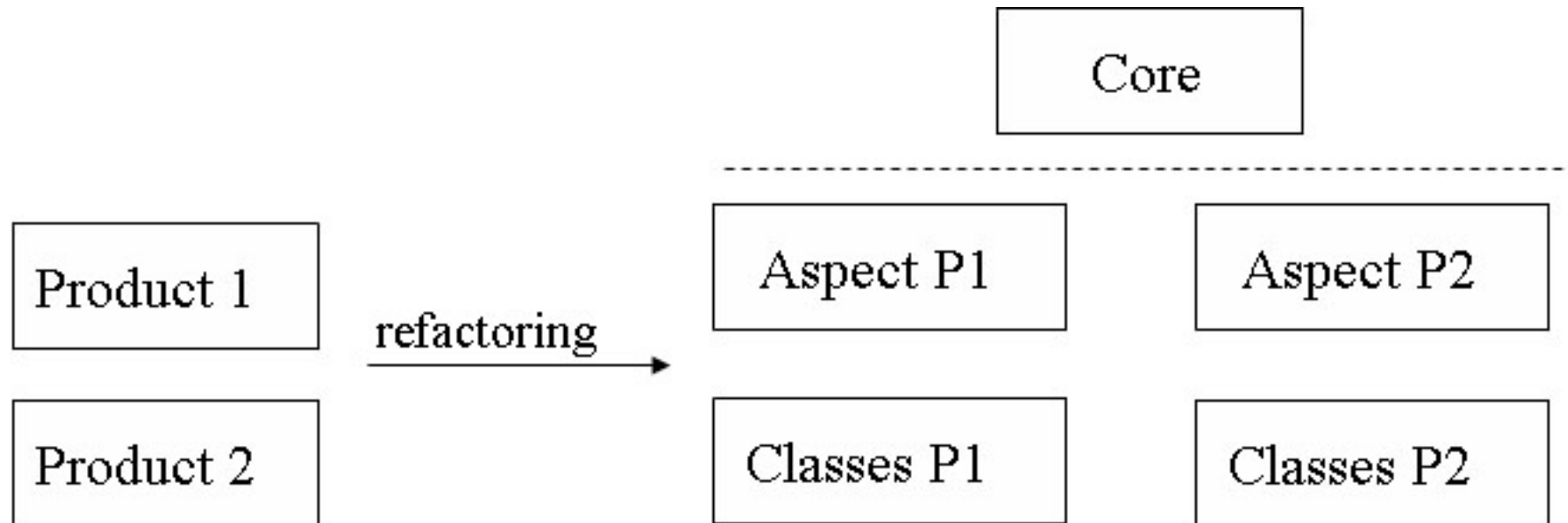
Besides the refactorings, we need ...

methods

to create and evolve product lines, combining adoption strategies for different

scenarios

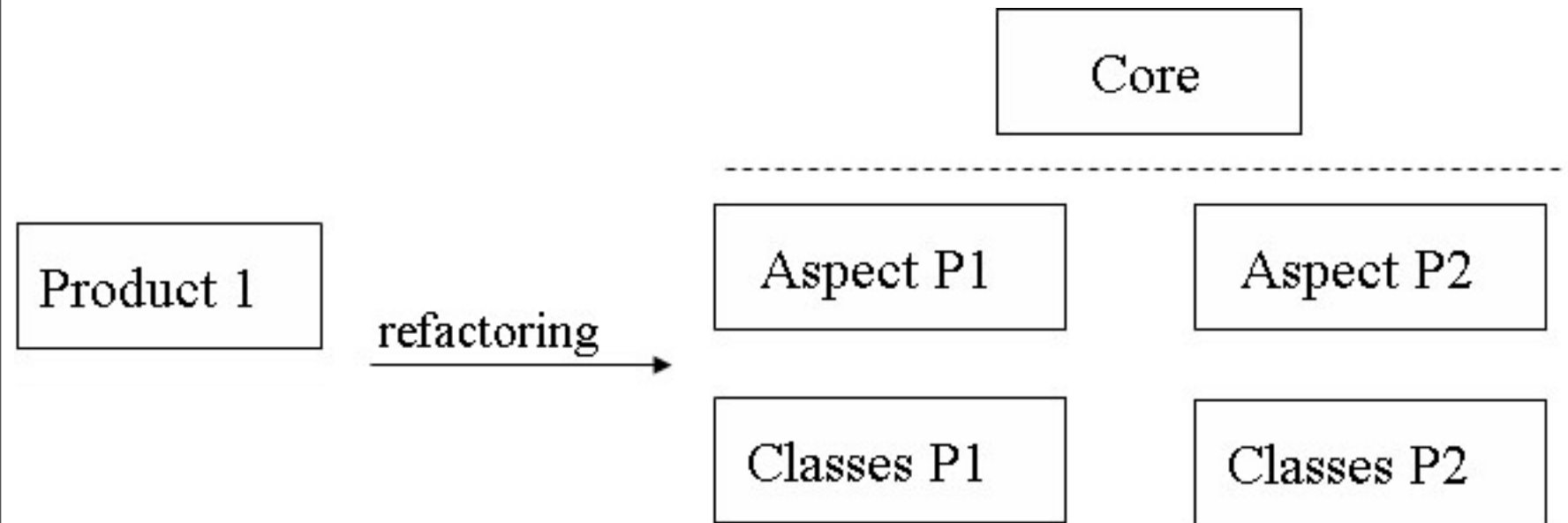
SPL extraction



Product1 = {Core, Classes P1 } ● Aspect P1

Product2 = {Core, Classes P2 } ● Aspect P2

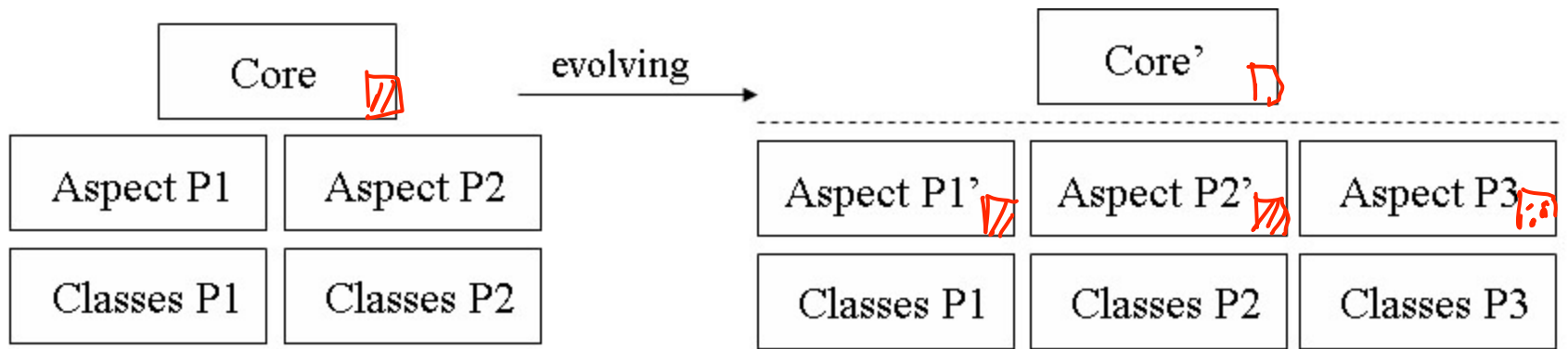
SPL derivation, porting



Product1 = {Core, Classes P1 } ● Aspect P1

Product2 = {Core, Classes P2 } ● Aspect P2

SPL evolution, new variation point



A3

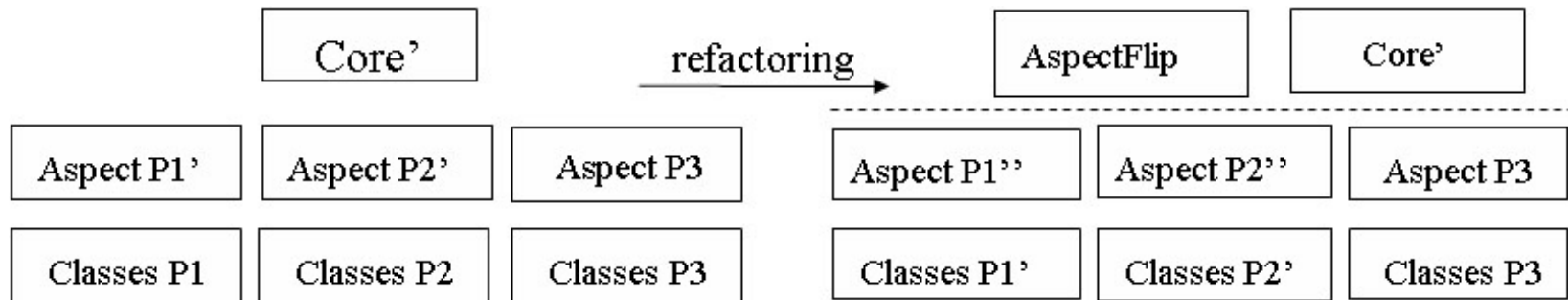
c3

Product1 = {Core', Classes P1 } • Aspect P1'

Product2 = {Core', Classes P2 } • Aspect P2'

Product3 = {Core', Classes P3 } • Aspect P3

SPL evolution, improvement

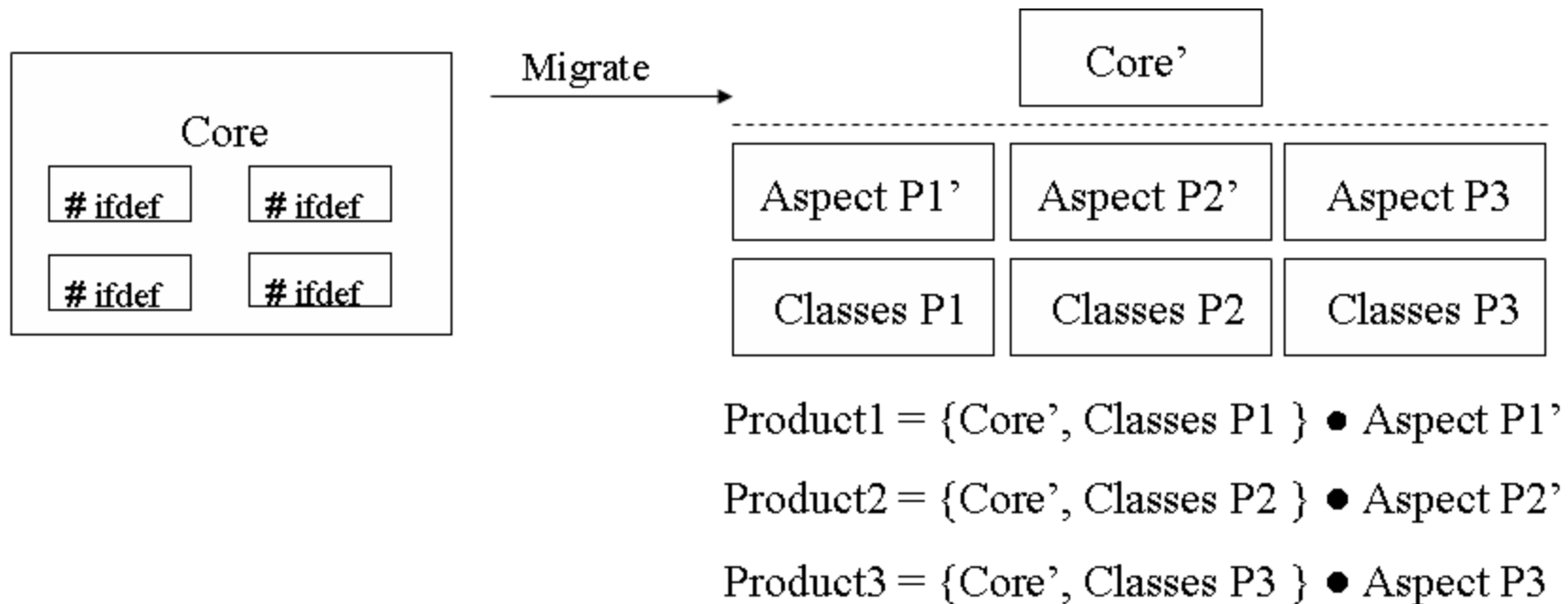


Product1 = {Core', Classes P1' } • {Aspect P1'', AspectFlip}

Product2 = {Core', Classes P2' } • {Aspect P2'', AspectFlip}

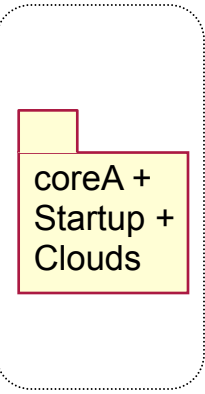
Product3 = {Core', Classes P3 } • Aspect P3

SPL migration



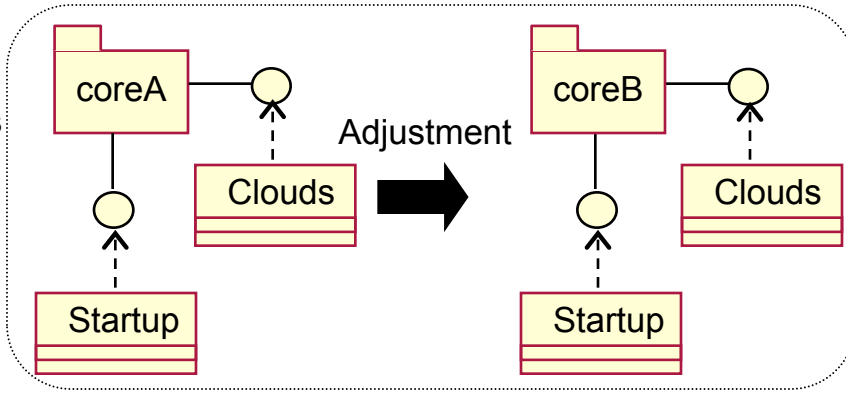
Case study: code refactoring

Product 1

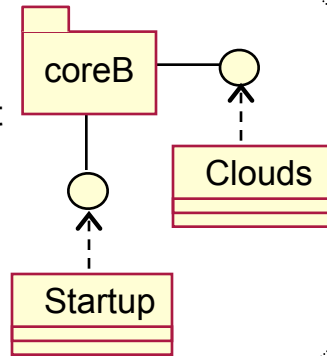


OA Refactorings

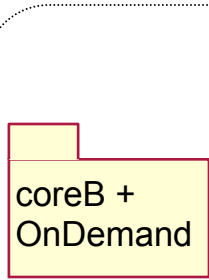
SPL 1



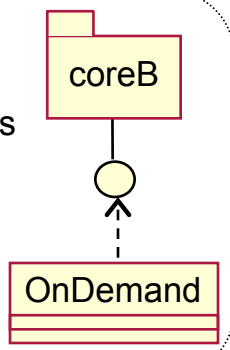
Adjustment



Product 2

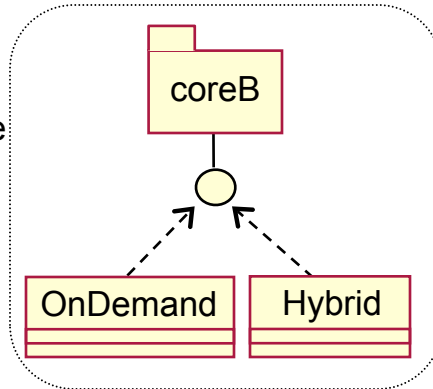


OA Refactorings



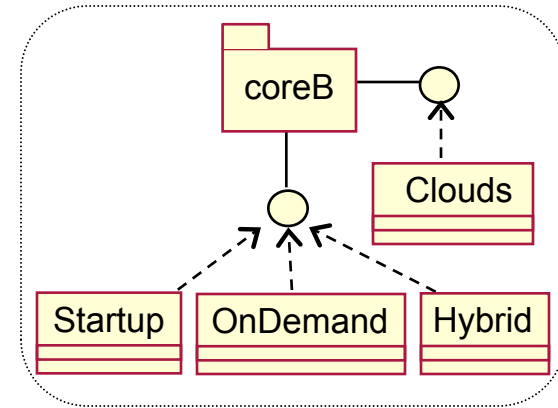
Reactive

SPL 2



→

SPL 1-2



Incremental method

- O núcleo (core) pode ser modificado
- Aspectos podem ser revisitados
- Componentes reusáveis são expostos
- Descrição do domínio construída iterativamente

Benefícios

- Sem "super-planejamento"
- Iterativo, assistido por ferramenta
- Núcleo separado das variações
- Simples configuração

Dificuldades e desvantagens

- Evolução do núcleo normalmente gera evolução dos aspectos
- Granularidade precisa de ajuste (refactoring)
- Template latente de código não capturado
- Localização/representação de features

Software product lines

- Benefits
 - Higher productivity
 - Higher quality
 - Low cost
- But higher complexity
 - Variability management
 - Adoption strategies

Practices and processes for software product lines

Paulo Borba

Informatics Center

Federal University of Pernambuco