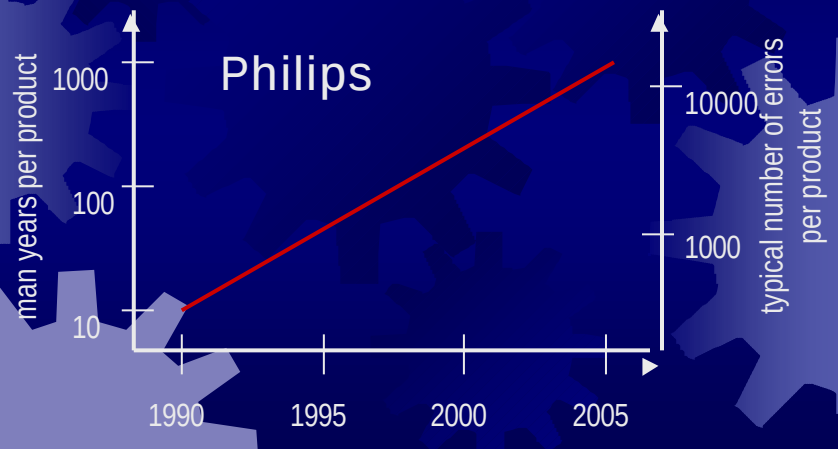
Management, Business and Organization in Software Product Lines

Jan Bosch
Professor of Software Engineering
University of Groningen, Netherlands
Jan.Bosch@cs.rug.nl
http://www.cs.rug.nl/~bosch

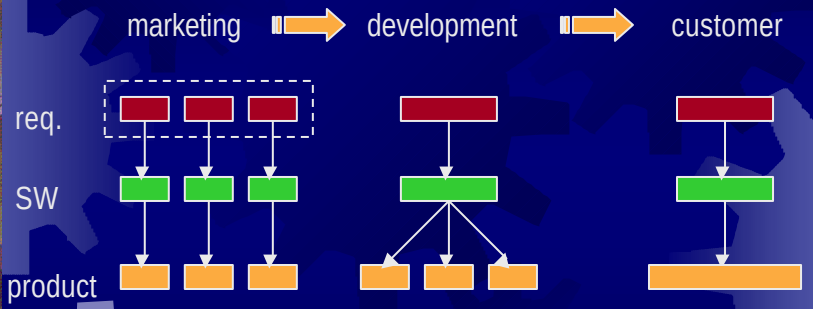
Software Product Line Workshop
ICSE 2002, May 21, Orlando

Trend: software size

- software needs in products constantly increasing



Trend: More Variability



post fielding upgrades
license key driven conf.
3rd part software

Software reuse

increasing SW size per product

increasing variability requirements

develop fewer products

develop more products

software product lines

Business Case Analysis

- why adopt architecture-centric software reuse?
  - decrease cost
    - software development
    - software maintenance
  - time to market
    - of new products
    - of new features in existing products
  - staff
    - unable to recruit new staff
    - need to maintain existing staff

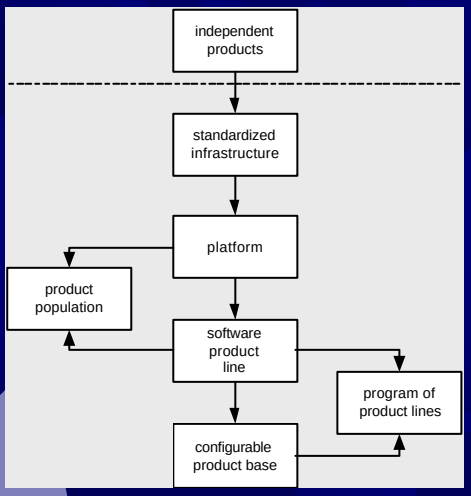
Transition to Software Product Lines

- traditional literature:
  - one technology approach
  - one business approach
  - one organizational model
- however, does “one size fit all”?

“Maturity” Levels

- standardized infrastructure
- platform (all common functionality)
- software product line
- configurable/generated product

“Maturity” Levels



“Maturity” Levels (contd.)

program of software product lines	product populations	configurable product
increase product size (hierarchical SPLs)	increase PL scope	decrease derivation cost
Nokia telecom switches (Philips Medical)	Philips consumer electronics	Telelarm

Standardized Infrastructure

- standardized infrastructure
  - operating system
  - database
  - GUI
  - etc.

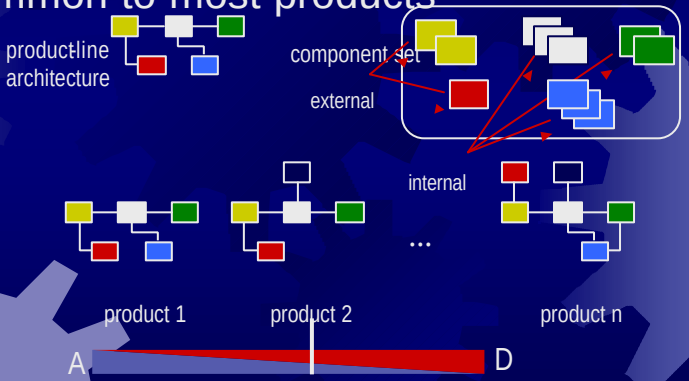
hardly any domain engineering activities for infrastructure evolution

Platform

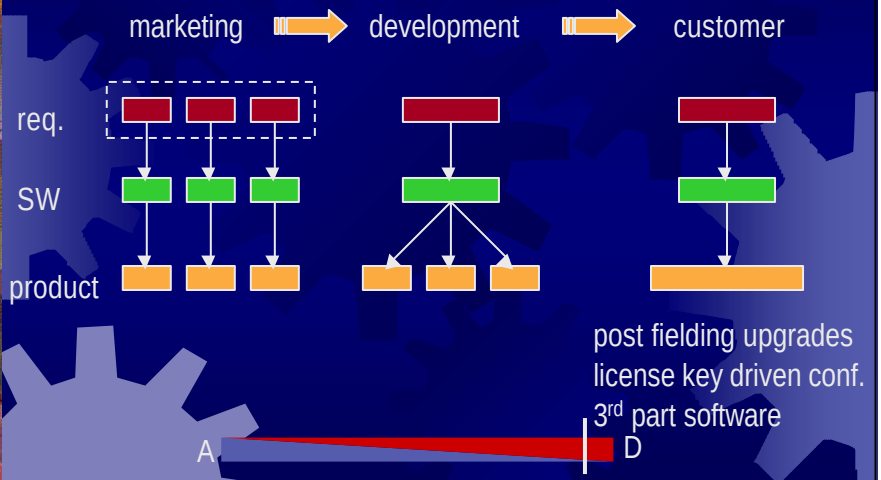
- functionality common to all products in the product line is captured
- both domain engineering and product engineering activities

Software Product Line

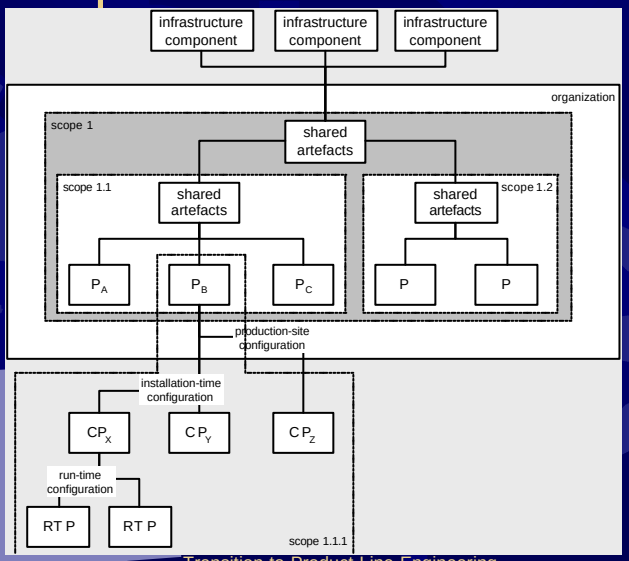
- domain assets capture functionality common to most products



Configurable Product



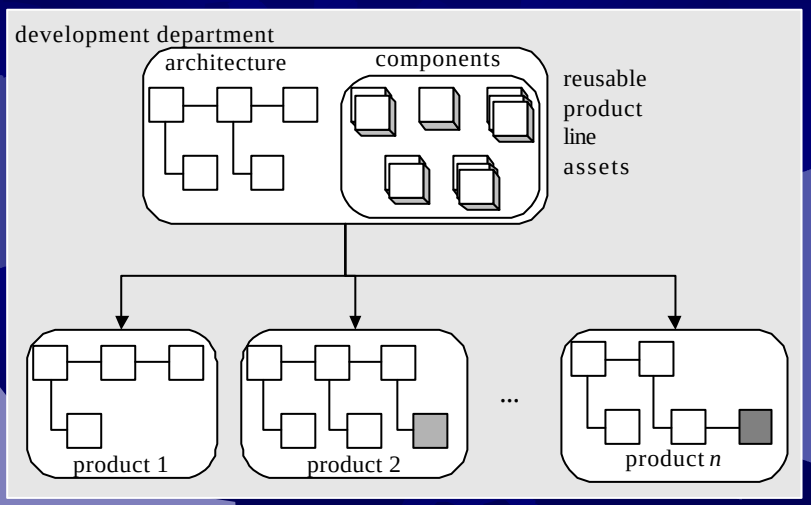
Example



Organizational Models

- development department
- business units
- domain engineering units
- hierarchical domain engineering units

Development Department



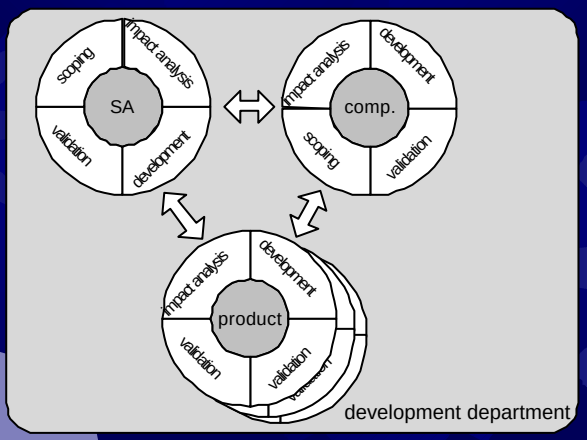
Development Department

applicability
- relatively small organizations (< 30)
- project rather product focus

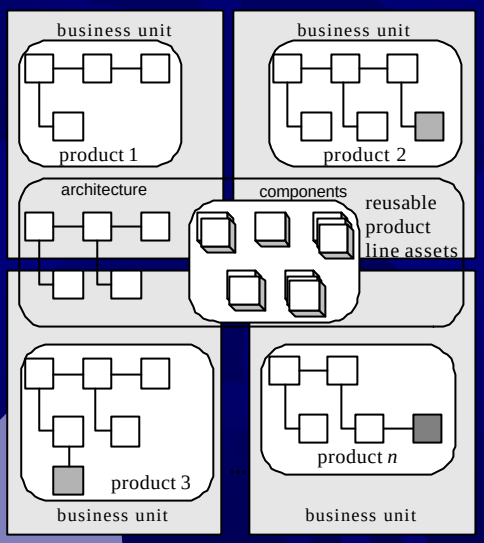
advantages
- simplicity
- ease of communication

disadvantages
- not scaleable
- focus on domain or application engineering

Development Department



Business Units



Business Units

three models

- unconstrained model
- asset responsibles
- mixed responsibility

conflicts

- origin of product line

Business Units

applicability

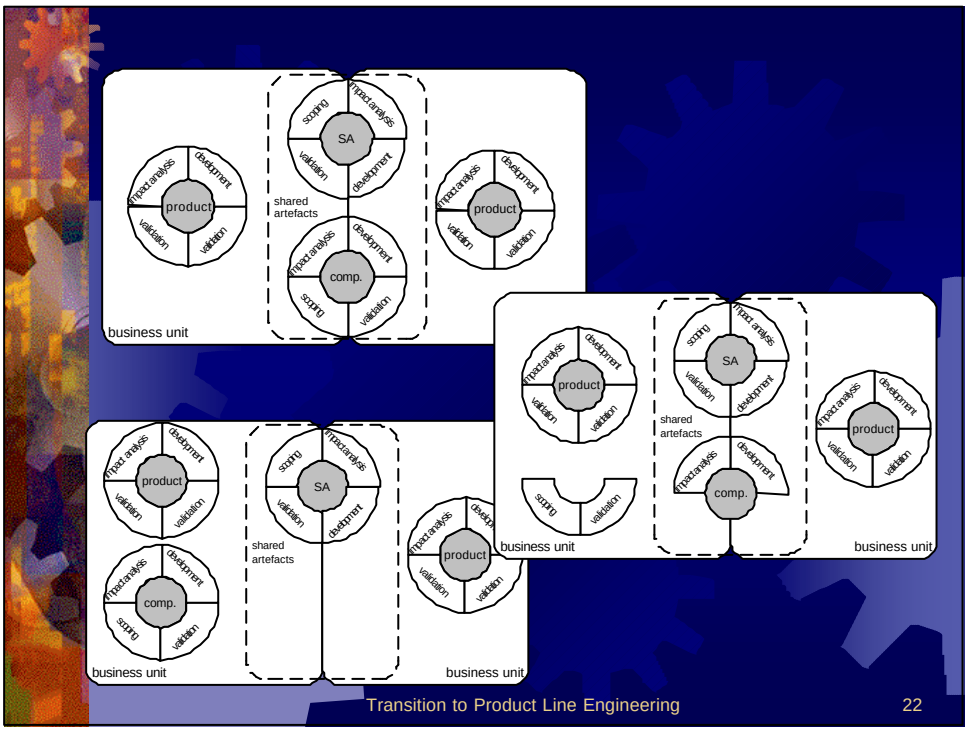
- size: 30 < org. size < 100

advantages

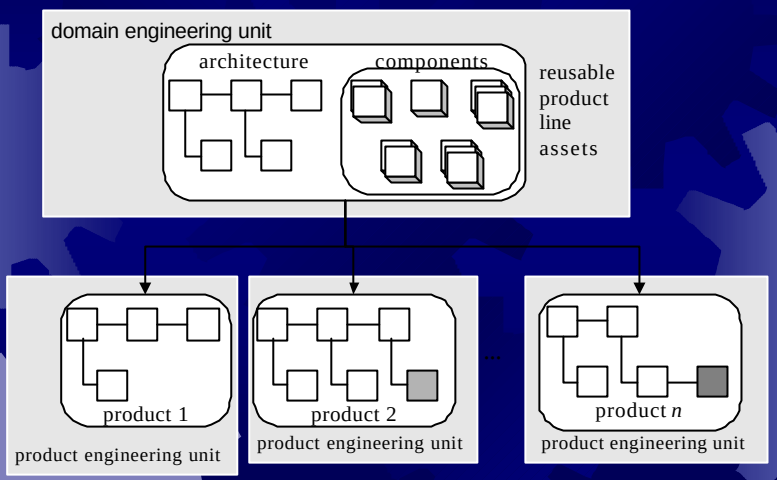
- effective sharing of assets (evolution!)
- better scaleability

disadvantage

- lack of attention to domain assets



Domain Engineering Unit



Domain Engineering Unit

two models

- single domain engineering unit
- one software architecture unit + component units

applicability

- size > 100 SEs but less than ??

Domain Engineering Unit

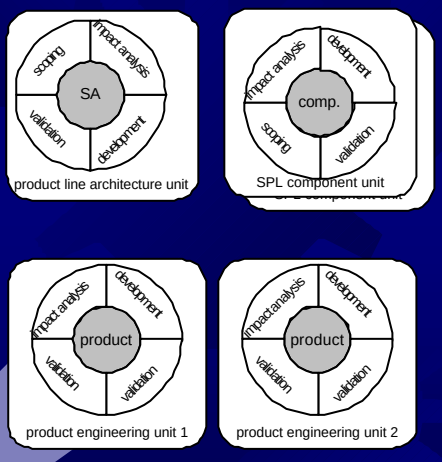
advantages

- reduces n-to-n communication to one-to-n
- focus on both domain engineering and product engineering
- even better scalability

disadvantages

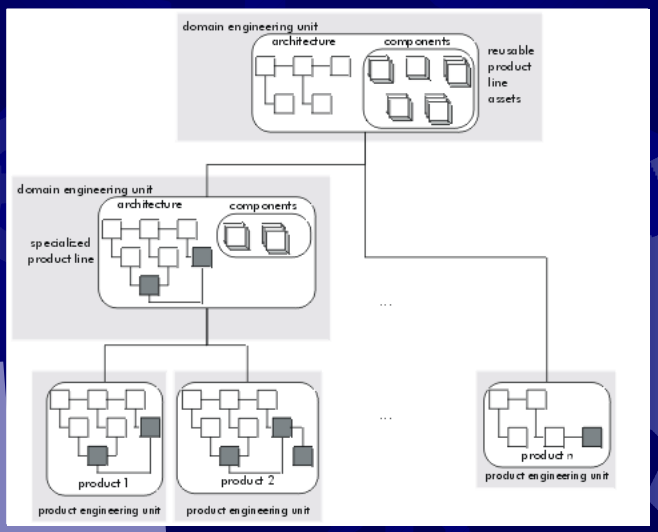
- complex management of evolution & information exchange DU and PU
- time to market of new features

Domain Engineering Unit



architecture and component units

Hierarchical DE Units



Hierarchical DE Units

applicability

- large or very large size (hundreds)

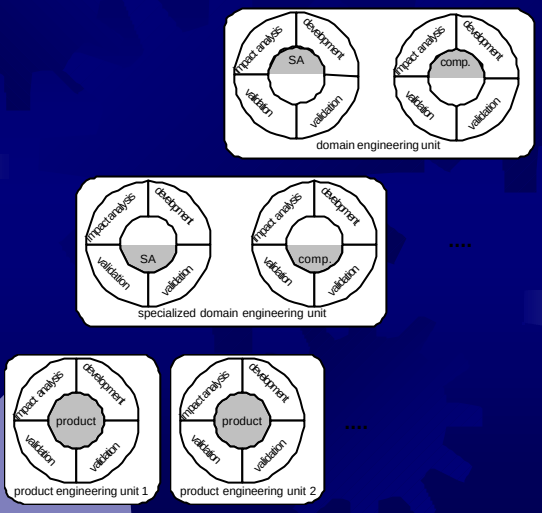
advantages

- manages large, complex systems and large staff

disadvantages

- organisational overhead

Hierarchical DE Units



Questions and Topics

- light-weight product line approaches
- product line adoption
  - quality improvement processes
  - focus on characterization of the organization
- multiple scopes – how to explicitly model