Model-Base Product Line Engineering – Variations on a Theme

Matthew Hause – Atego Chief Consulting Engineer

Agenda

- System Complexity and Variability
- Product Lines
- Model-Based Systems Engineering
- Model Asset Reuse
- Model-Based Product Line Engineering
- Conclusions
- Questions?

System Complexity and Variability

- Automotive systems are complex
  - Thousands of requirements
  - Multiple systems containing multiple processors
  - Multiple variants and variability
  - Often driven by customer need
- Variability can affect multiple systems
  - Interconnections and interdependencies difficult to manage
  - Mutual exclusion of parts
  - Commonality between systems
  - Achieving economies of scale
- A new approach is needed

3

Product Lines

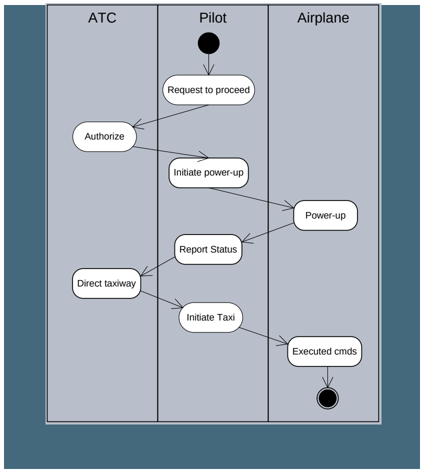
- In existence since the Industrial Revolution
- Common in automotive industry
  - Henry Ford was one of the first
  - Assembly lines with interchangeable parts
  - Often evolve over time rather than designed
  - Difficult to manage and control
- Typical Product Lines
  - ~80% of Products the Same with Predictable Component Costs
  - Commonality drives down costs
  - Focus on better Decision Making for the Variable ~20%

Changes in Systems Engineering Practice

Change from Document centric to Model centric

Old Approach

- Requirement Specifications
- Interface Definitions
- System Architecture
- System Functionality
- Trade-off Analysis
- Test Specifications
- Etc.

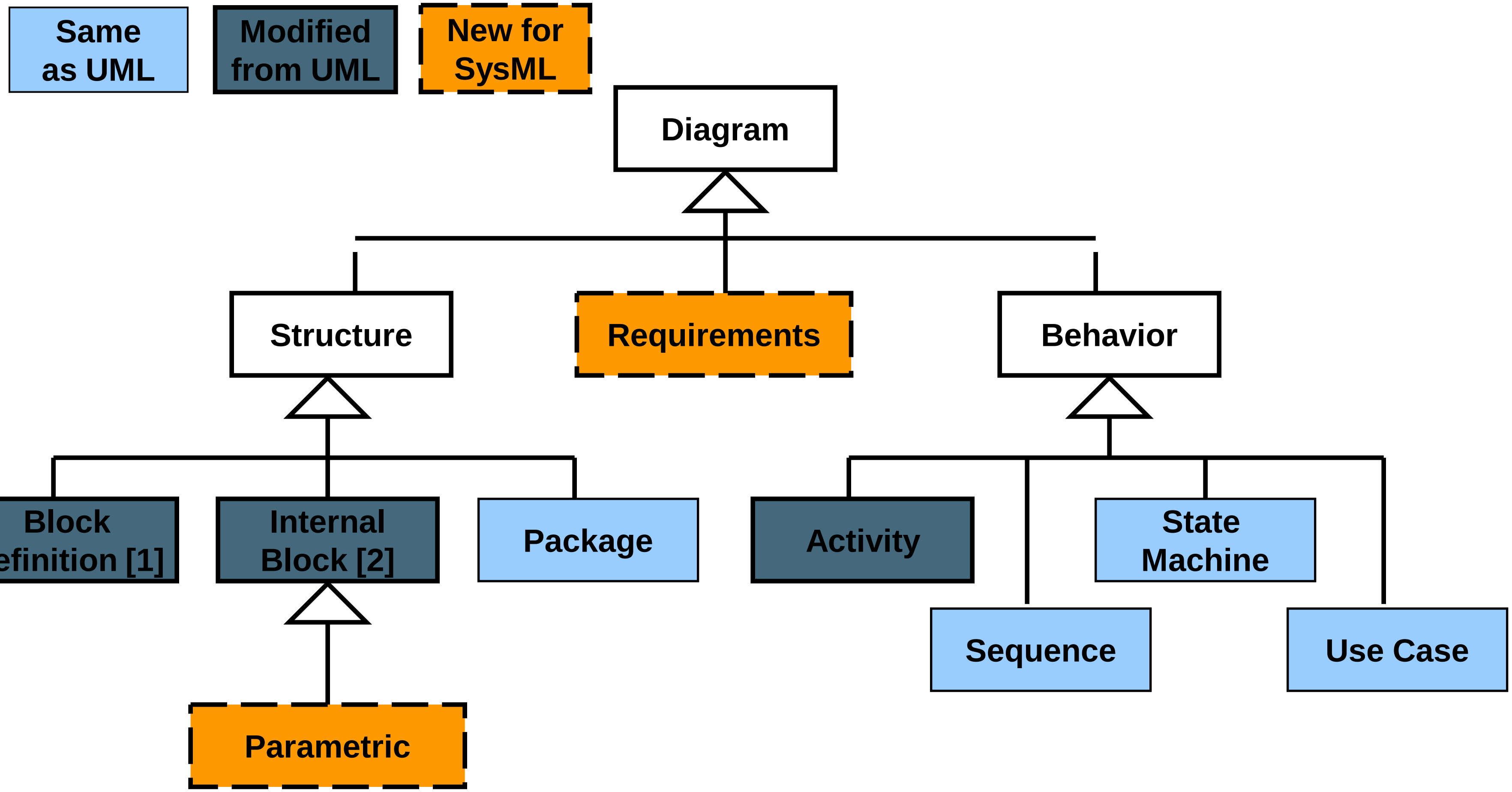


New Approach

Modeling – A Way to Manage Complexity

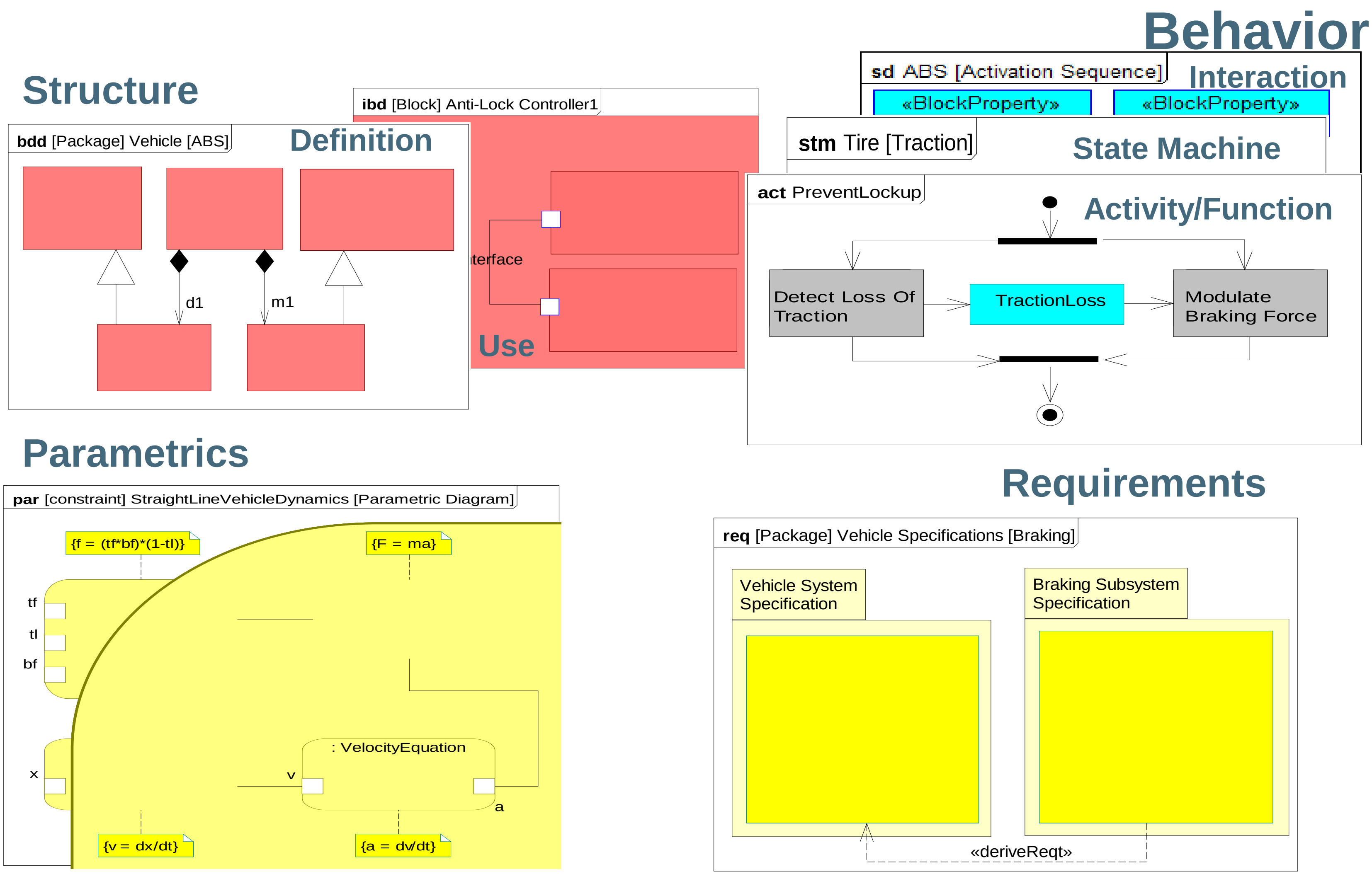
- **A Model is a set of *entities* and *relations* created as a result of an abstraction process and is used to:**
  - neglect immaterial details, hence foster communication and system understanding
  - anticipate the relevant system behavior
  - test and recycle, because it is an operational design tool
- **User can visualize entities and relations through *diagrams* which form the system's *views***
  - **Views are a set of information that describe a *partial* and *particular* system aspect: *the human brain can only take into account a few things all together at the same time***
- **A Database is needed to archive all the Model information**

SysML Taxonomy of Diagrams

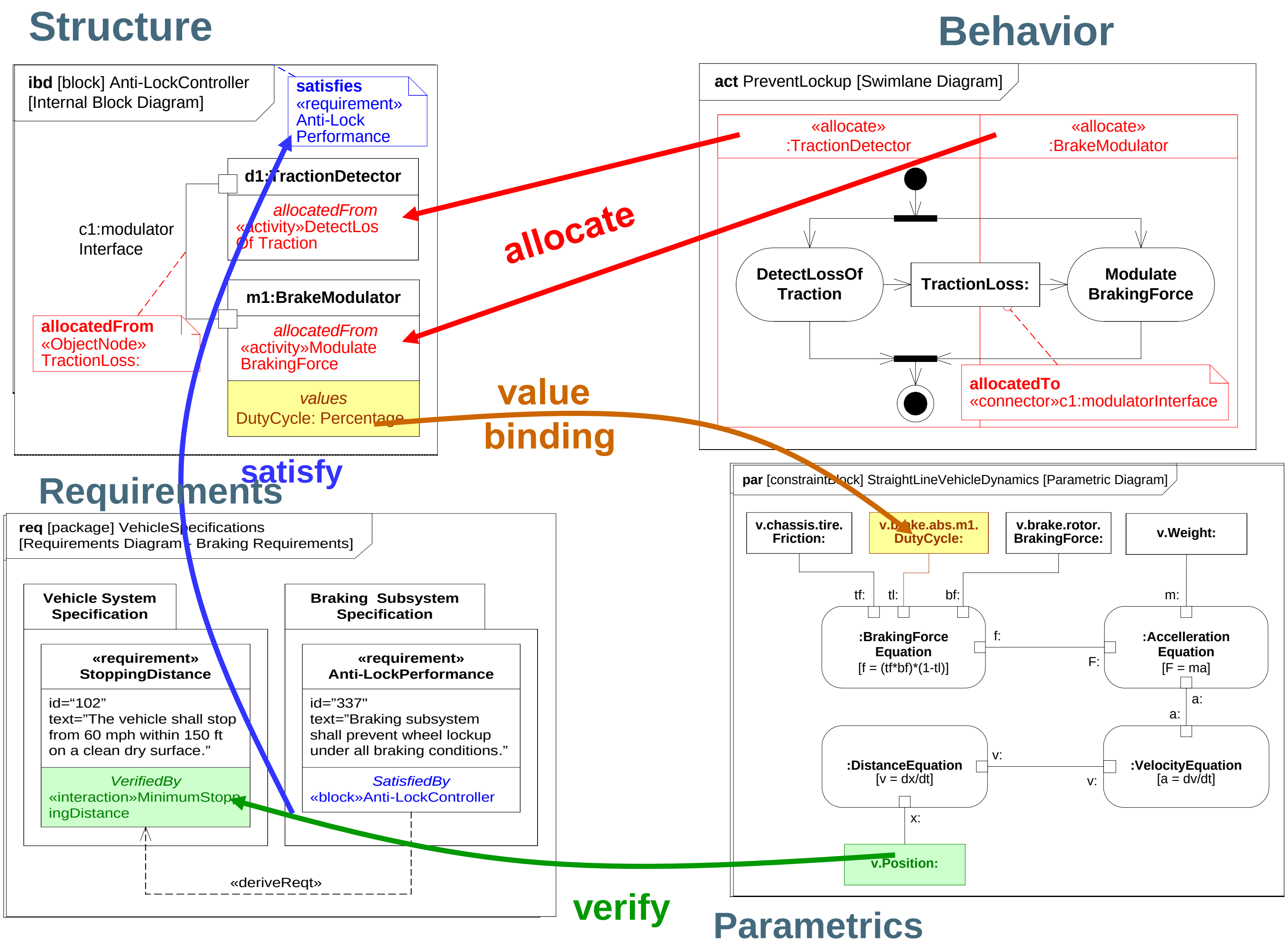


[1] Modified UML Class Diagram
[2] Enhanced UML Composite Structure Diagram

The Four Pillars of SysML

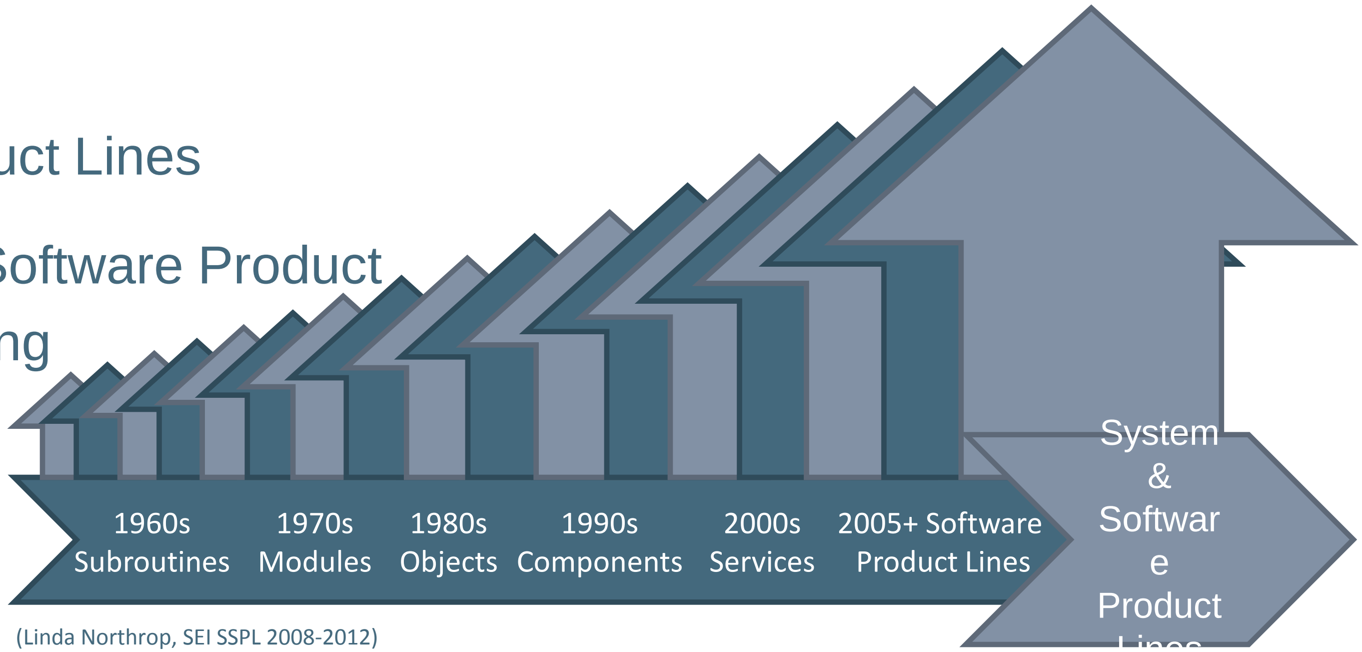


Cross Connecting Model Elements



Reuse Paradigms

- Subroutines
- Modules
- Objects
- Components
- Software Product Lines
- Systems and Software Product Line Engineering



(Linda Northrop, SEI SSPL 2008-2012)

Model Asset Reuse

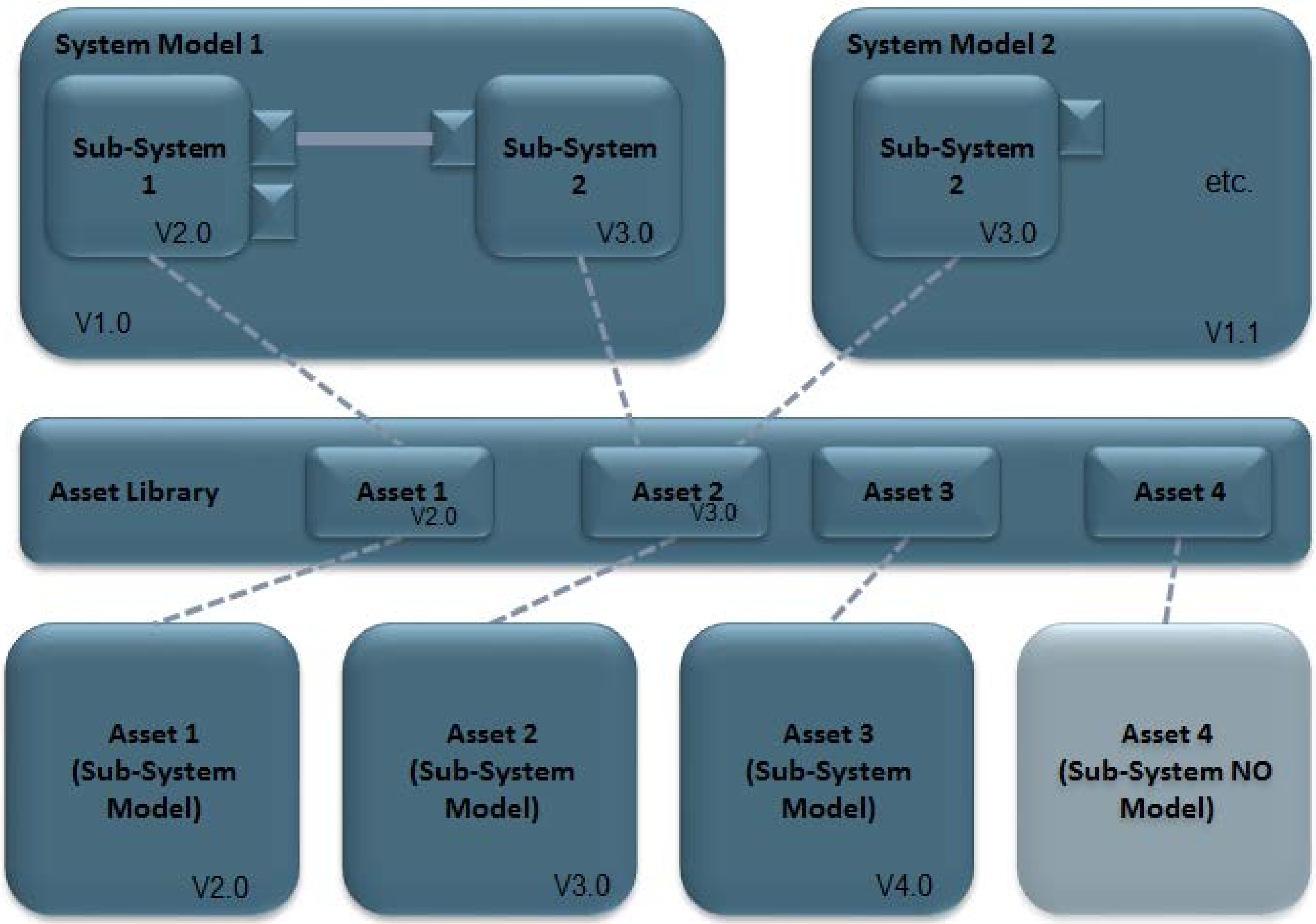
- The OMG Reusable Asset Specification (RAS)
  - Used for defining reusable assets, their interfaces, characteristics and supporting elements
- Three key dimensions describe reusable assets:
  - Granularity describes how many particular problems or solution alternatives a packaged asset addresses.
  - The visibility varies from black-box assets, whose internals cannot be seen and are not modifiable, to white box assets which are visible and modifiable.
  - The articulation describes the degree of completeness of the artifacts in providing the solution.
- Asset also include supporting documentation, requirements addressed, interfaces, etc.
- Provides a standards-based "model of models" approach instead of a "mega-model" approach.

Asset Reuse

Higher Level Models

Links via Assets

Lower Level Models



Variability Management is needed…

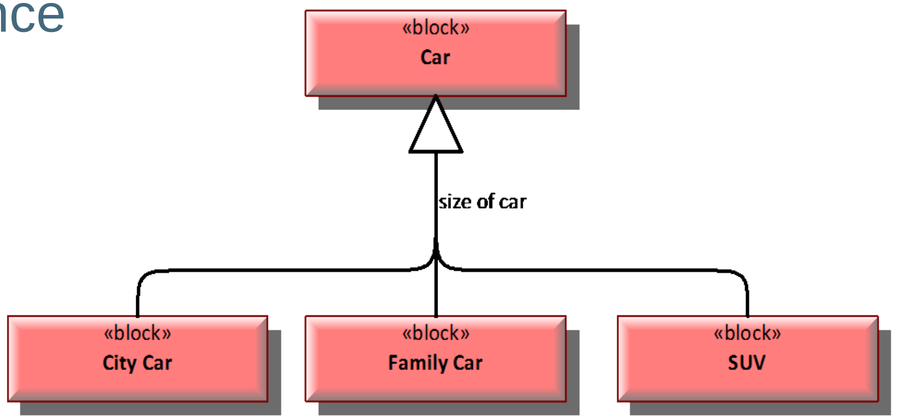
- In all phases of the development process:
- Requirements Engineering
  - What are common Requirements?
- System and Software Architecture
  - Different and/or Common Architectures?
- Implementation
  - How do the system and software differ in the products?
- Tests
  - Re-usable (common) test cases?

Product Line Modeling

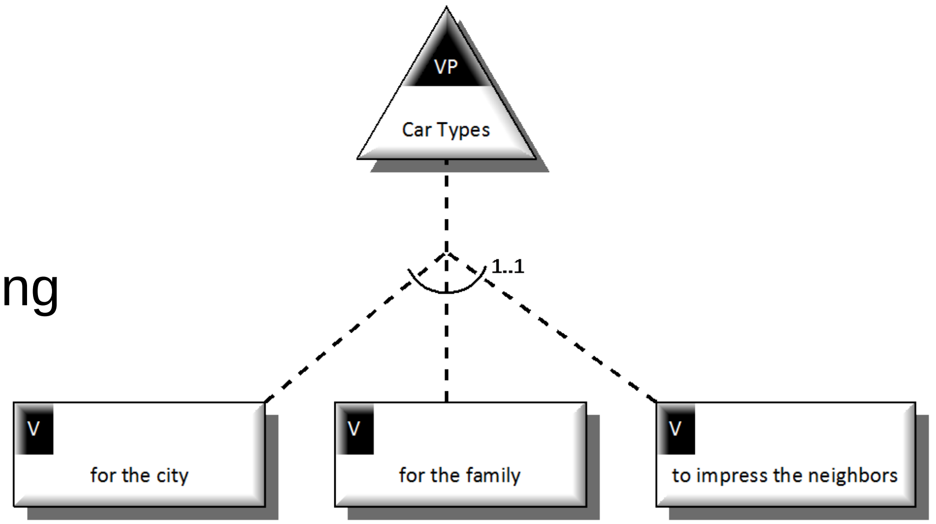
- Object Variability Modeling (OVM)
- The concept of ‘Variability’ Modelling in OVM
  - Variation Points
  - Variants
  - Variability Constraints
- Integrates variability modeling with systems modeling
- References:
  - ISO26550:2013 – Reference Model for System and Software Product Line Engineering and Management
  - Klaus Pohl, Günter Böckle, Frank van der Linden, Software Product Line Engineering – Foundations, Principles, and Techniques, 2005

Variability Approaches

- Model Variability using inheritance



- Model Variability using OVM
  – Orthogonal Variability Modeling



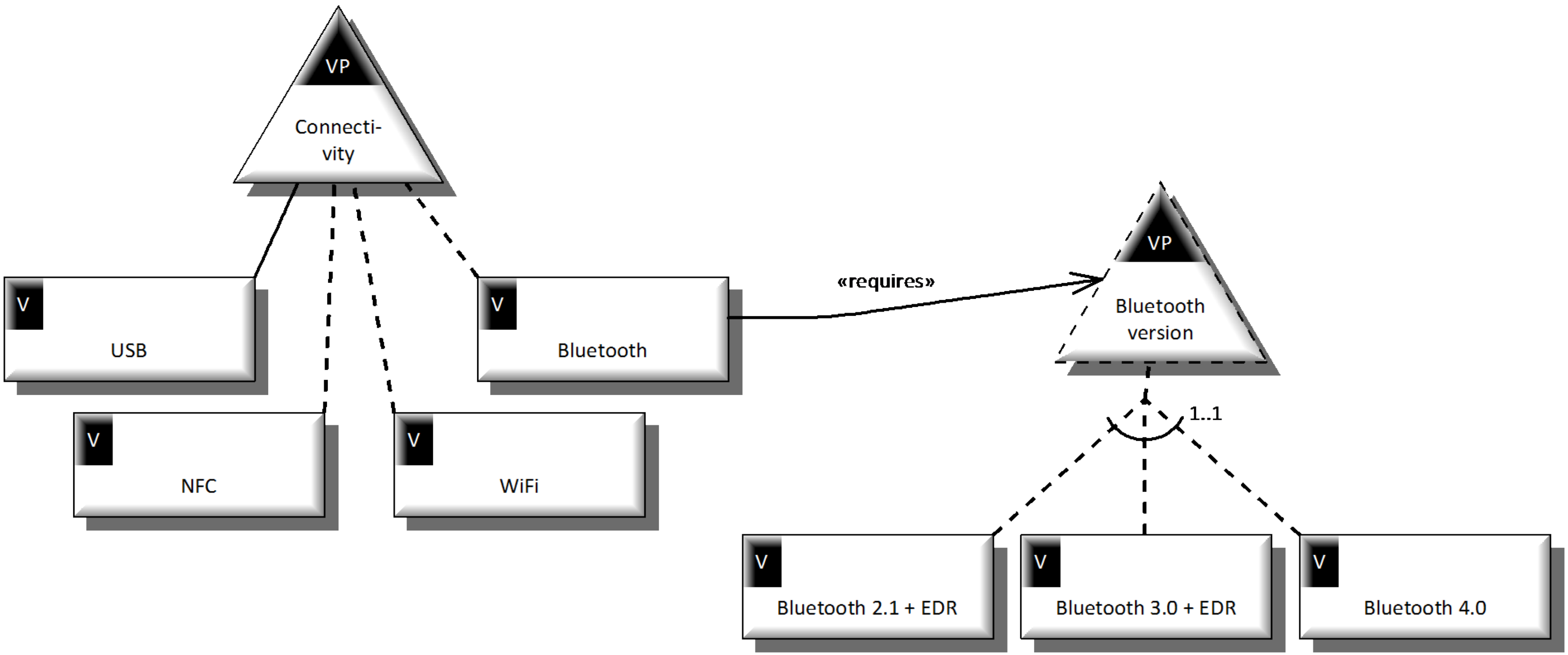
Model-based Variability using OVM

- Allows both approaches
  - Separate views for variability
  - Link of variable elements with base model elements
  - Shared views showing variability and base model views
- Combination of both views within one model enables to
  - Separate the concerns
  - Keep base and variability models consistent
- Model-based variable Assets
  - As an option to combine product families with component-based design

atego

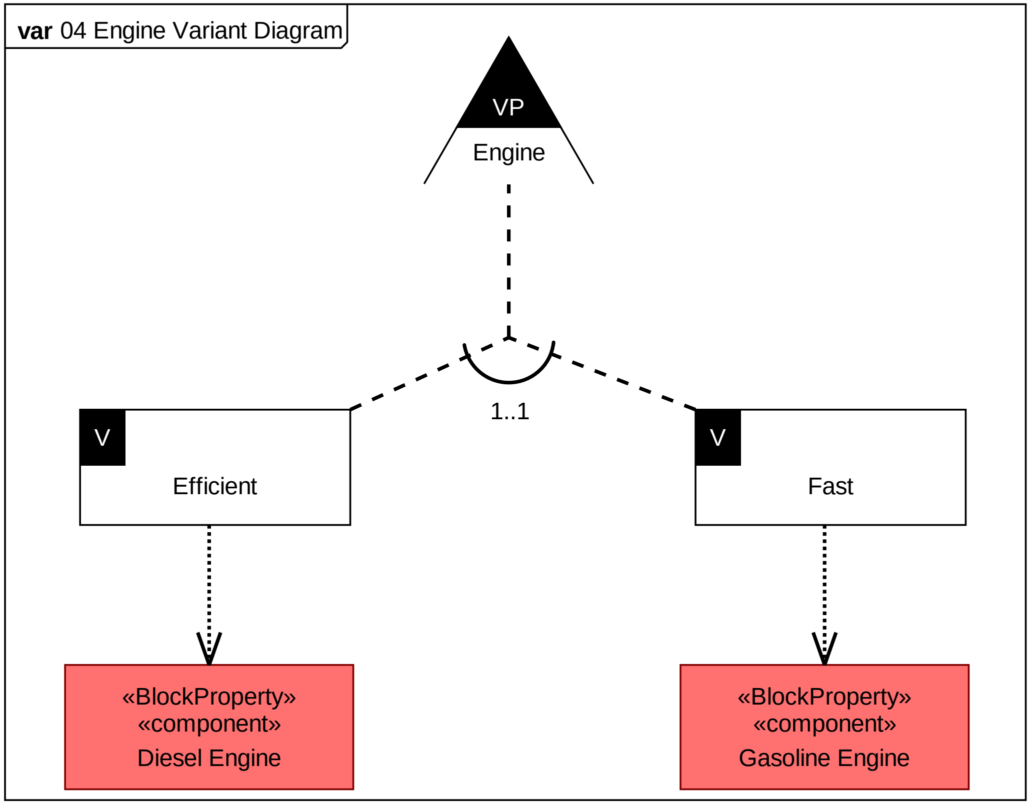
Modeling Variability

- The selection of the Bluetooth connectivity requires the selection of the Bluetooth version



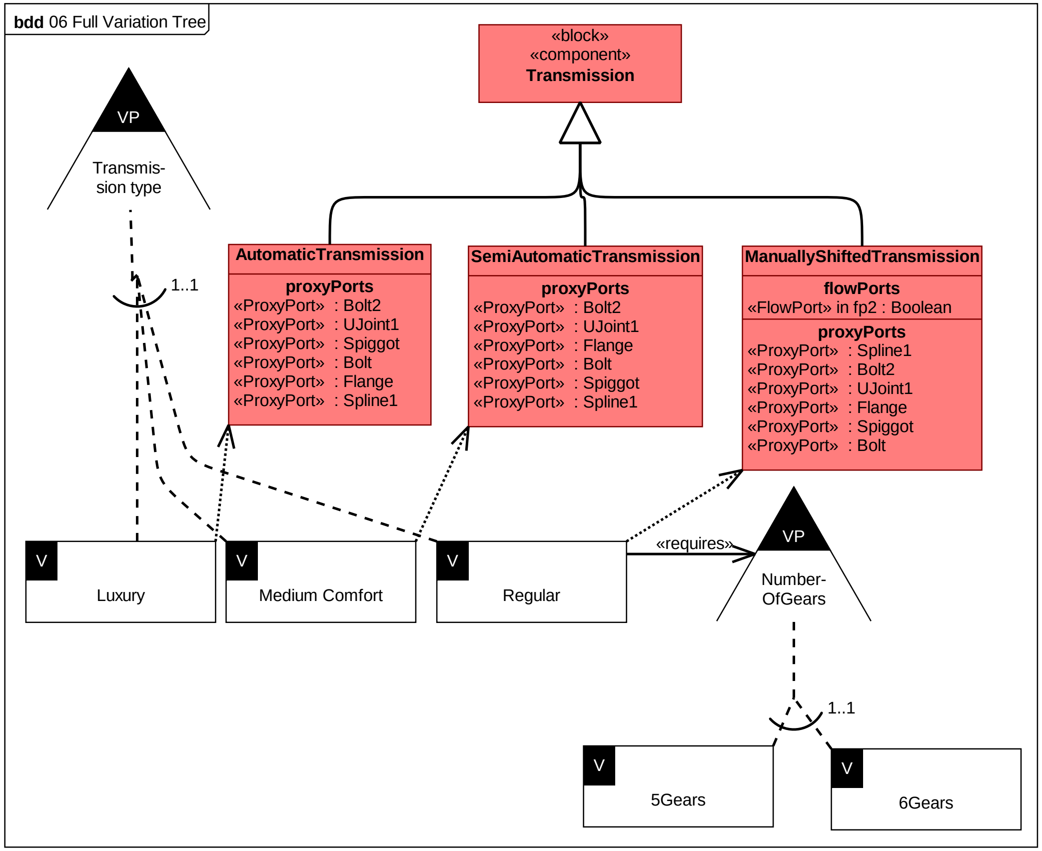
Modeling Variability

- Engine variability along with model dependencies



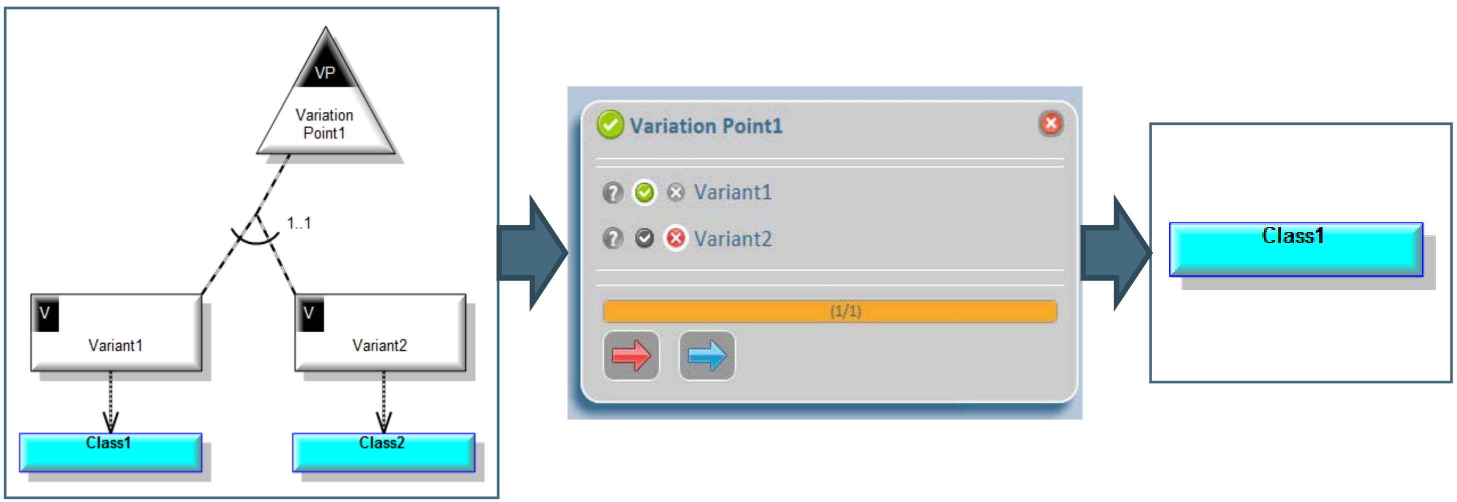
Modeling Variability

- Types of Transmission



Product Model Creation

- Auto-Creates Product Models
  - Variability Decisions Applied
  - Unnecessary Variation Points, Variants & Base Model Artefacts Removed



- New Product Model Branch, Original Product Line Model Retained
- Product Model suitable for Trade Studies, Simulation & Generation

Decision Set Editor Example

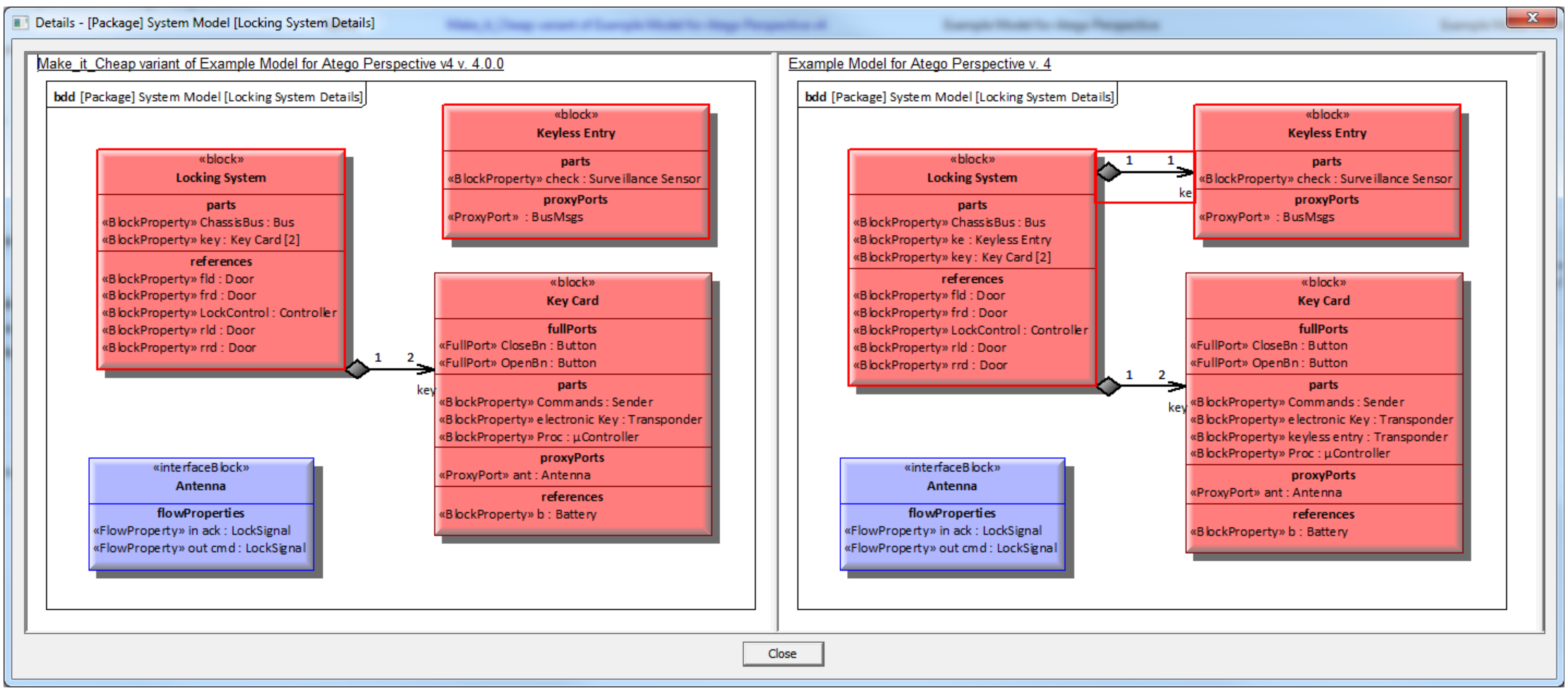
●○		Name	Decision	Status	Included By	Excluded By	Reason
		Dynamic	?	Excluded		Variable sequence diagrams::Variability.Keyless Entry, Variable sequence diagrams::Variability.Line.Alternative Choice1	
		Luxus	?	Excluded		Variable sequence diagrams::Variability.Keyless Entry, Variable sequence diagrams::Variability.Line.Alternative Choice1	
		Keyless Entry Option	?	Excluded		Variable sequence diagrams::Variability.Base	
M	E	Line		Included	Variable sequence diagrams::Variability.Base		
		Base	Include	Included			
		Dynamic	?	Excluded		Variable sequence diagrams::Variability.Keyless Entry, Variable sequence diagrams::Variability.Line.Alternative Choice1	
		Luxus	?	Excluded		Variable sequence diagrams::Variability.Keyless Entry, Variable sequence diagrams::Variability.Line.Alternative Choice1	
M		Multimedia Supplier		Included	Variable sequence diagrams::Variability.The cheaper, better, but quite unknown Brand		
		The well-known, expensive Brand	?	Excluded		Variable sequence diagrams::Variability.Multimedia Supplier.Alternative Choice1	
		The cheaper, better, but quite unknown Brand	Include	Included			
M		Battery Size		Included	Variable sequence diagrams::Variability.44Ah		
		65Ah	?	Excluded		Variable sequence diagrams::Variability.Base	
		44Ah	?	Included	Variable sequence diagrams::Variability.Battery Size.Alternative Choice1		

Variants: 16/3ː | Variation Points: 8/21 | Inconsistent: 0 | Undecided Mandatory Variation Points: 9

Variant Selector Example

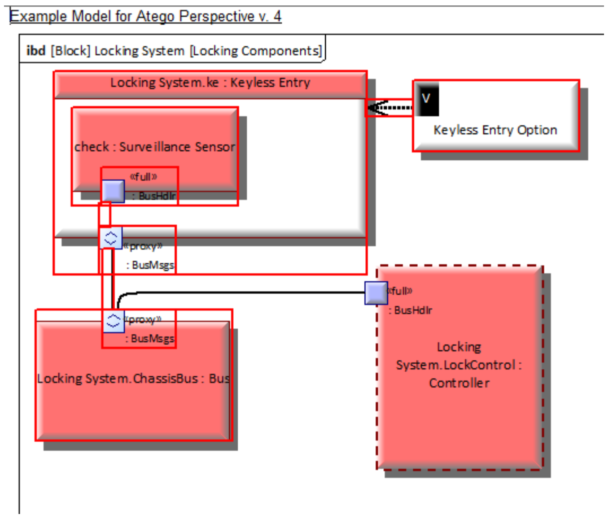
Pruned Elements Example in SysML BDD

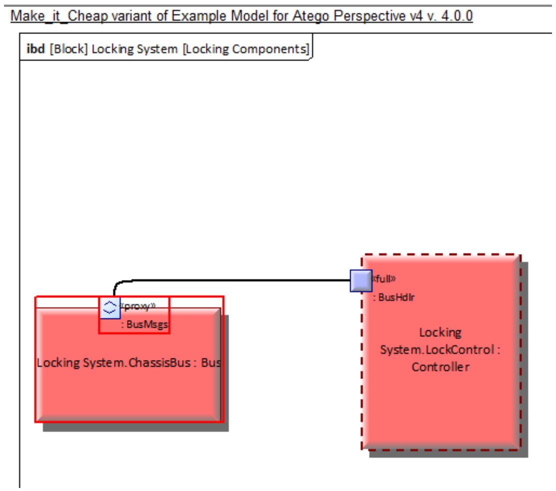
- Parts are deleted
  - ke : Keyless Entry in Locking System



Pruned Elements in SysML IBD

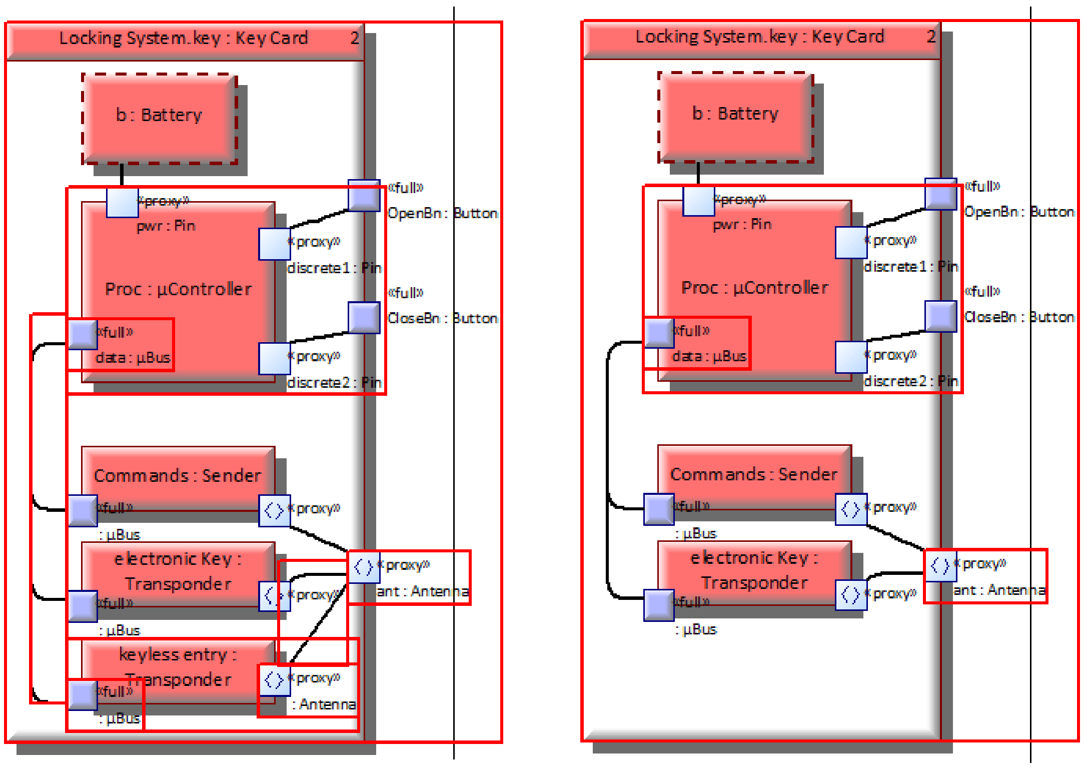
- The (unselected) Keyless Entry Variant is gone, including all Artifact Dependencies
- Keyless Entry Part is deleted from the Locking System
- All sub-parts, ports and connectors relating to the Keyless Entry Part are deleted as well





Pruned Elements in SysML IBD (cont‘d)

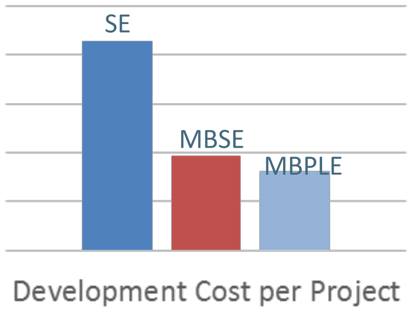
- Within the key:Key Card Part, the relevant Transponder is pruned
- All sub-parts, ports and connectors relating to this Transponder sub-part are deleted as well

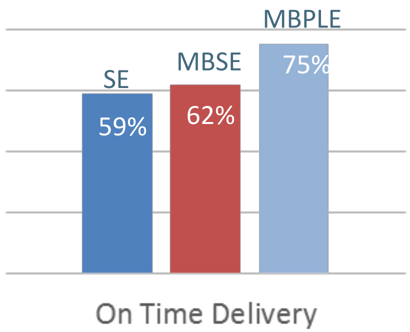


atego

Cost Reduction & Time Improvements

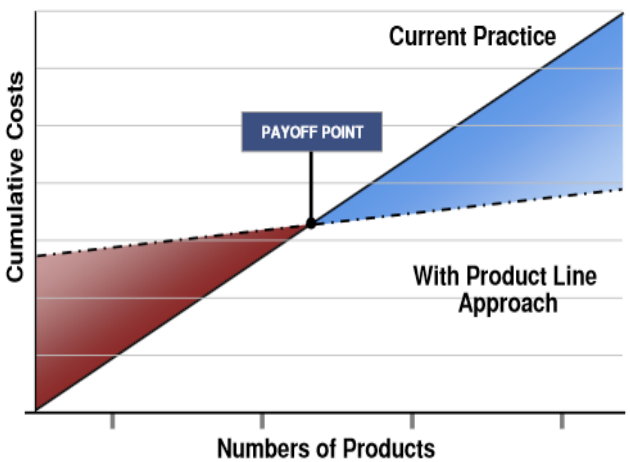
- SE (Non-Modelled Systems Engineering)
  - 59% of Projects Delivered on Time
- MBSE (Model Based Systems Engineering)
  - 62% of Projects Delivered on Time
- Compared to SE
  - 55% Reduction in Total Development Cost per Project
  - 16% More Project Delivered on Time
- MB-PLE (Model Based Product Line Engineering)
  - 75% of Projects Delivered on Time
- Compared to MBSE
  - 17% Reduction in Total Development Cost per Project
  - 6% More Projects Delivered on Time
- Compared to SE
  - 62% Reduction in Total Development Cost per Project
  - 23% More Projects Delivered on Time





(EMF 2013 Independent Survey Results from 667 Systems engineering respondents)

Benefits Summary



Weiss. D.M. & and Lai, C.T.R..
Software Product-Line Engineering: A Family-Based Software Development Process
Reading, MA: Addison-Wesley, 1999.

- SEI Carnegie Mellon University
  - Improved productivity by as much as 10x
  - Increased quality by as much as 10x
  - Decreased cost by as much as 60%
  - Decreased labor needs by as much as 87%
  - Decreased time to market by as much as 98%
  - Ability to move into new markets in months, not years

(Linda Northrop, SEI SSPL 2008-2012)

Conclusions

- Systems and models of systems have become increasingly complex.
- New ways are needed to organize models and the decisions made while creating them.
- SysML, Product Line Engineering, the Object Variability Modeling and the Reusable Asset Specification provide Model-Based Product Line Engineering (MB-PLE).
- Enables the asset reuse while making value-based decisions on system configuration.
- Provides a demonstrable ROI that will reduce development time and costs and help automotive engineers build better systems.

28

Questions and Answers