SysML for Telescope System Modeling
- Variant Modeling -

by the

INCOSE MBSE Challenge Team SE^2

Speaker

Tim Weilkiens, managing director of the German consultancy oose GmbH, is a member of INCOSE MBSE Challenge Team SE^2 (Telescope modeling). He is also an active member of the OMG working groups about SysML and UML and has written sections of the SysML specification.

Rainer Diekmann, works as an independent consultant in the aerospace industry, joins the INCOSE MBSE Challenge Team SE^2 (Telescope modeling) a year ago. He work(ed)s in system engineering projects focussing on MBSE, using different modelling approaches like UML, SysML and IDEF.

Agenda

- **What is the SE^2 Challenge project about?**
- Overview Variant Modeling
- The SYSMOD Variant Profile for SysML
- Variant Configurations
- Managing Complexity – Building simple views
- What's next?

About SE^2

- Collaboration between European Southern Observatory (ESO) and German Chapter of INCOSE (GfSE)
- Access to high-tech project, the Active Phasing Experiment (APE).
- The team members are:
  - Robert Karban (ESO)
  - Tim Weilkiens (oose GmbH)
  - Rudolf Hauber (HOOD Group)
  - Rainer Diekmann
  - Michele Zamparelli (ESO)
  - Andreas Hein (TU Munich)
- Former members: Andreas Peukert (TU Munich)

ESO

Non-profit Intergovernmental European Organisation for Astronomical Research in the Southern Hemisphere

http://www.eso.org

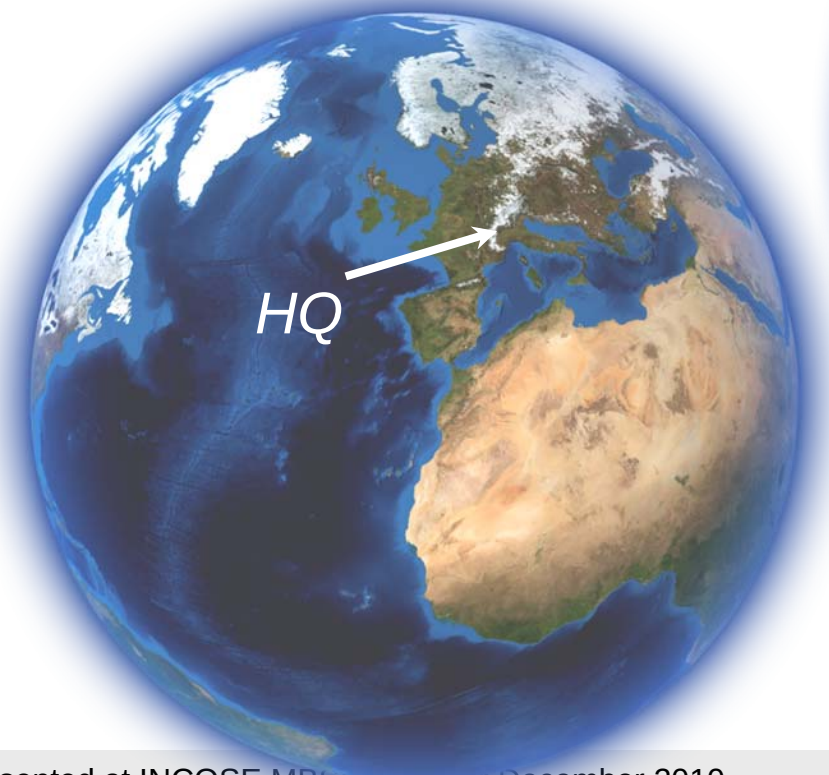
Headquarters in Munich, Germany, 3 Observatories in Chile

Mission statement

Build and operate world class ground based astronomical facilities

ESO's sites

- Paranal (2600 m)
- La Silla (2400 m)
- Chajnantor (5000 m)



ESO major projects

Very Large Telescope (VLT)

Started 1988, in operation since 1999

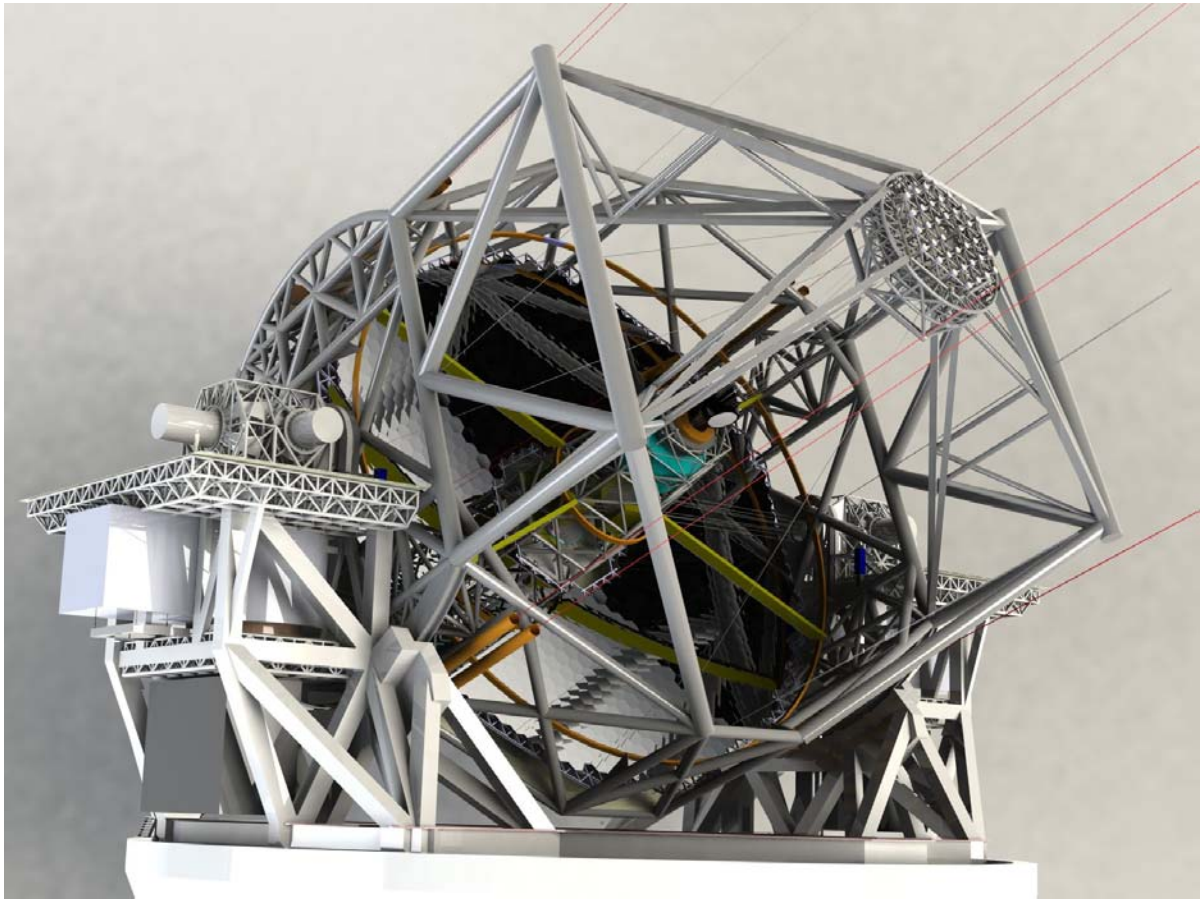
Atacama Large Millimeter Array (ALMA)

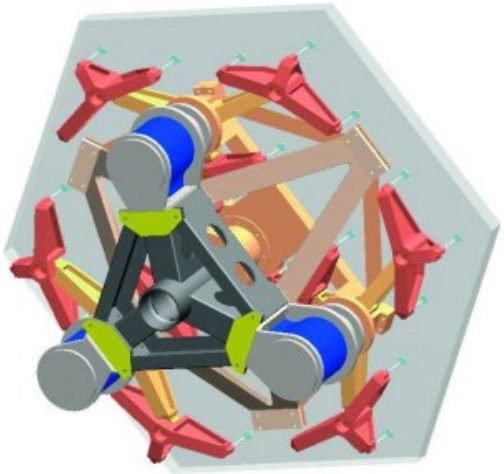
Europe-US-Japan

Started 1998, installation starting now

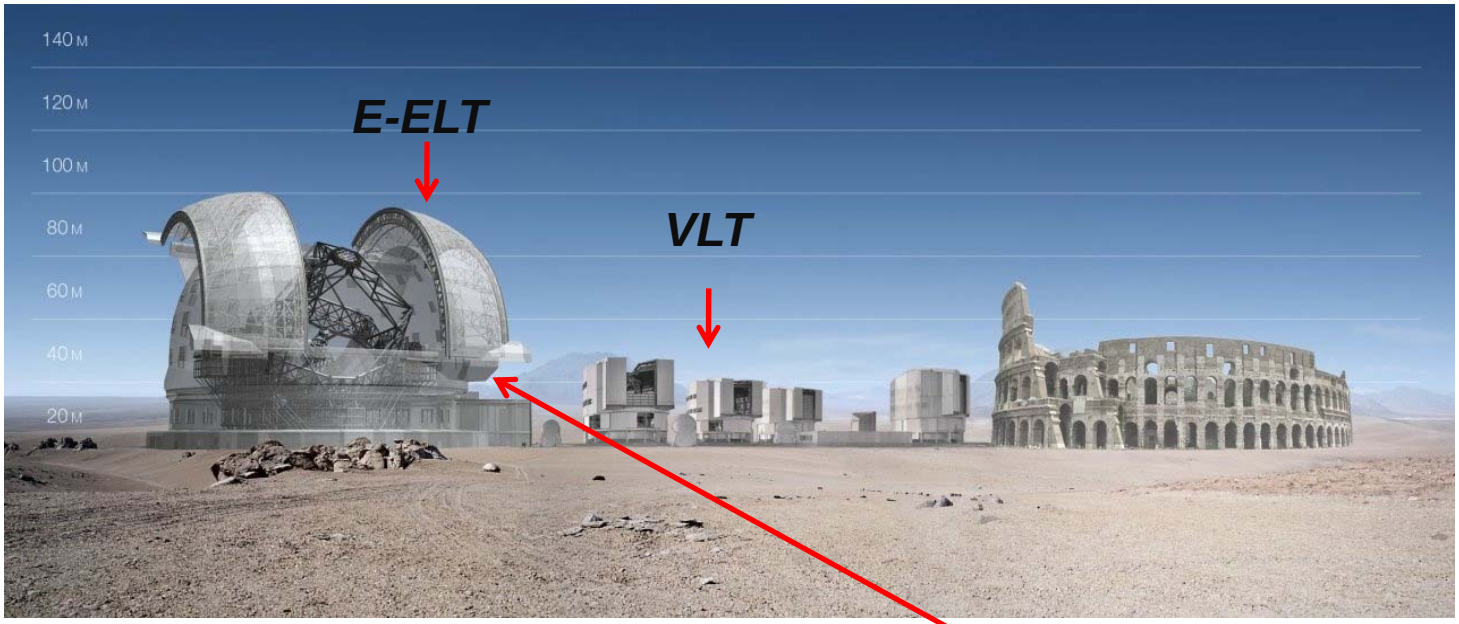
Images on this slide were produced by ESO

E-ELT

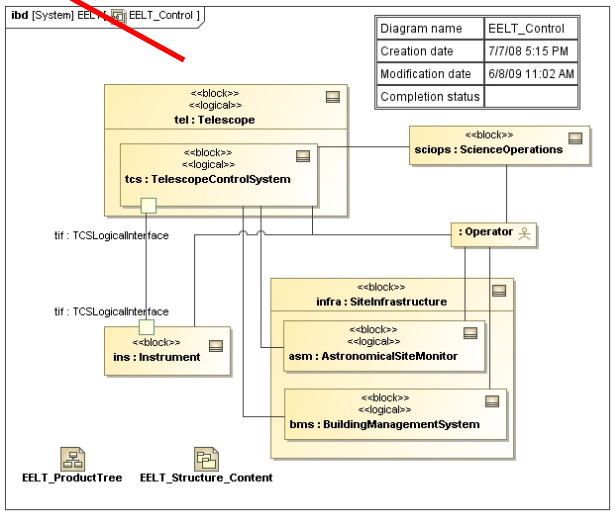




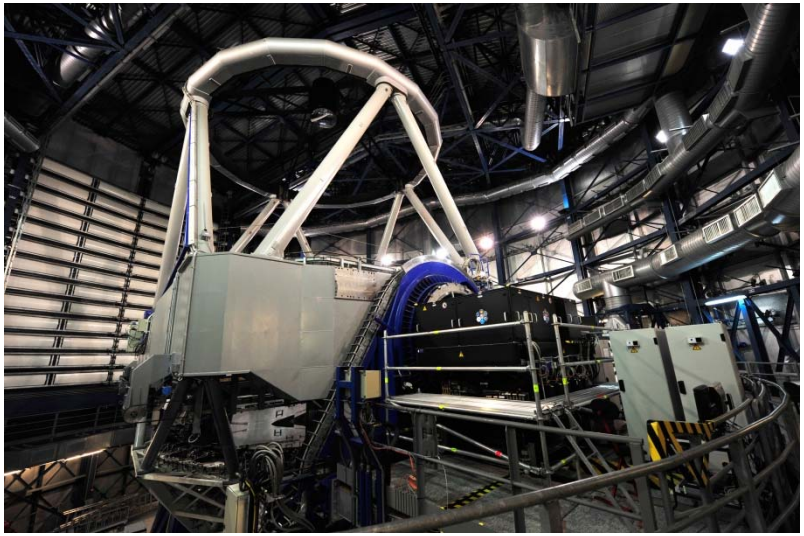
Images on this slide were produced by ESO



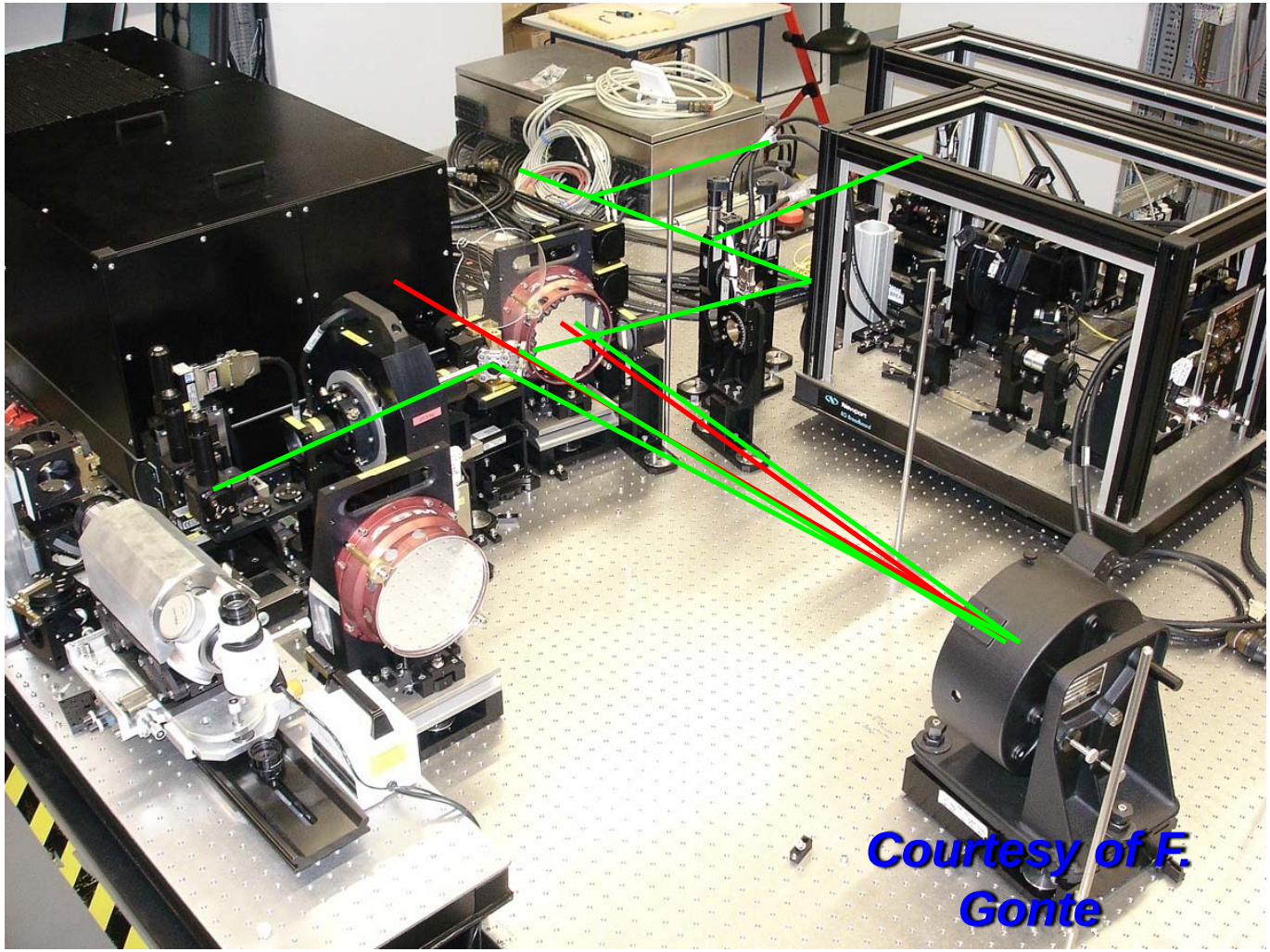
- 10000 tons of steel and glass
- 42m segmented primary mirror
- 20000 actuators, 1000 mirrors
- 50000 I/O points, 700Gflops/s, 17Gbyte/s
- Many distributed control loops
- Use MBSE/SysML to model the control system since 2008

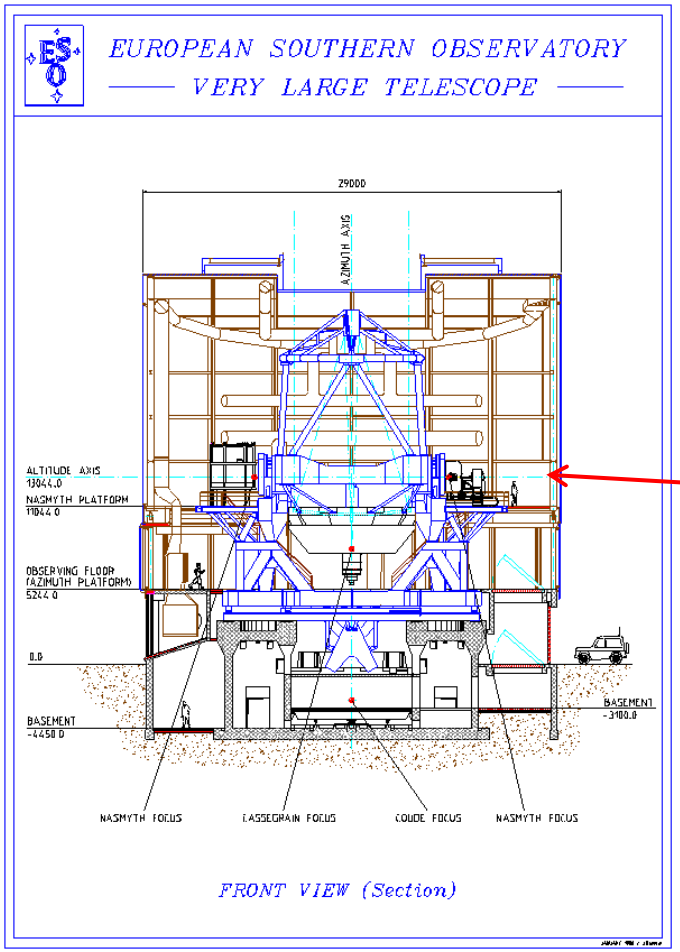


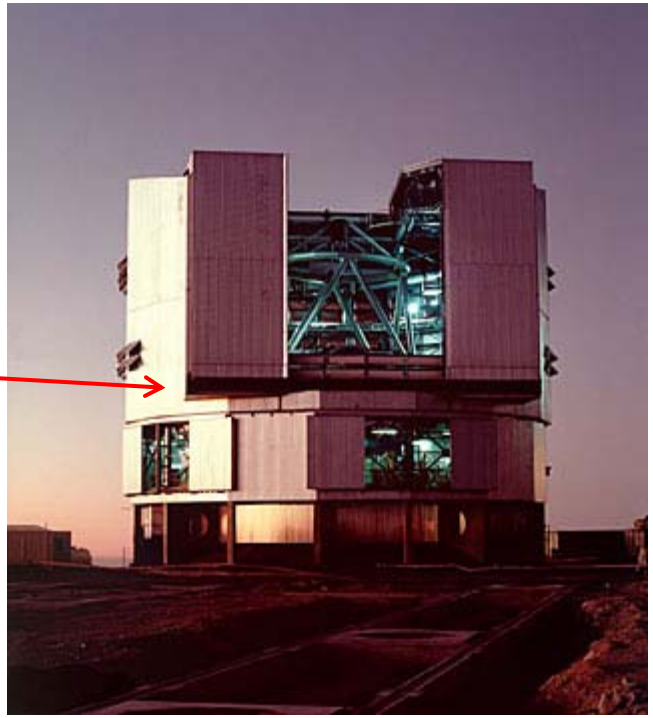
What is the challenge project about?



- System case study (since 2007)
  - APE technology demonstrator for future Extremely Large Telescope (ELT)
  - High-Tech interdisciplinary opto-mechatronical system in operation at Paranal observatory
- Goals
  - Create modeling guidelines and conventions for all system aspects, hierarchy levels, and views
  - Create fully fledged SysML model
  - **Documented at http://mbse.gfse.de**



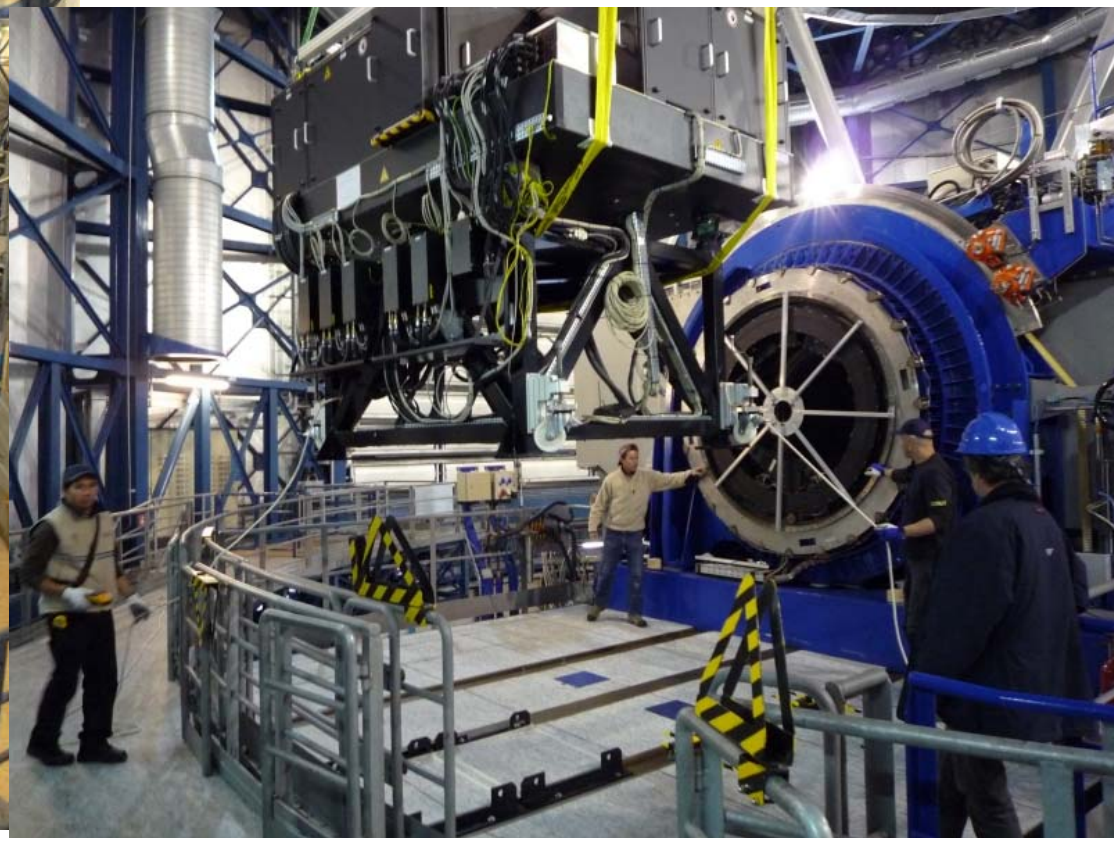




APE was installed at telescope in Atacama desert, Chile.

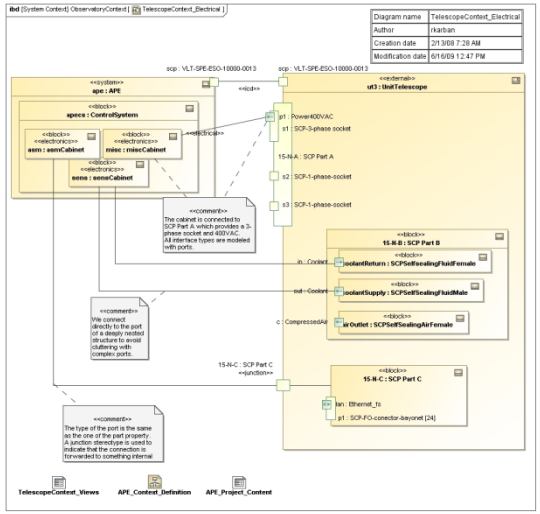
Images on this slide were produced by ESO

Installation on the platform of the telescope



What have we achieved?

- APE model, guidelines and best practices: MBSE Cookbook
  - Model structure and overview
  - Objectives and Requirements
  - Context, System Structure
  - Behavior and Data
  - Verification
  - Model library and SE Profile
- Plug-in for modeling tool
- Input for tool vendor and SysML RTF



Agenda

- What is the SE^2 Challenge project about?
- **Overview Variant Modeling**
- The SYSMOD Variant Profile for SysML
- Variant Configurations
- Managing Complexity – Building simple views
- What's next?

The value of variant modeling

The modeling of system variants is a core technique for model based systems engineering. You need to model variants

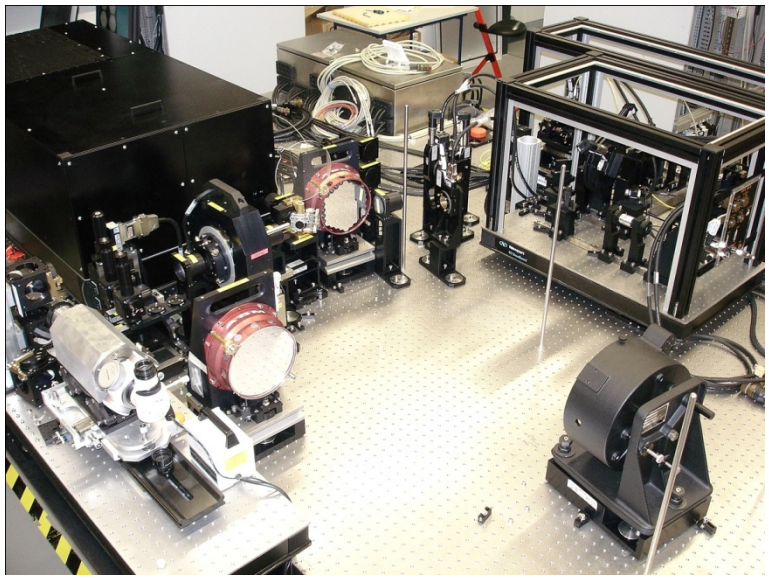
- for analysing design alternatives,
- for evaluating variants via trade-offs,
- for modeling of product families,
- and for the separation of a logical and a physical architecture.

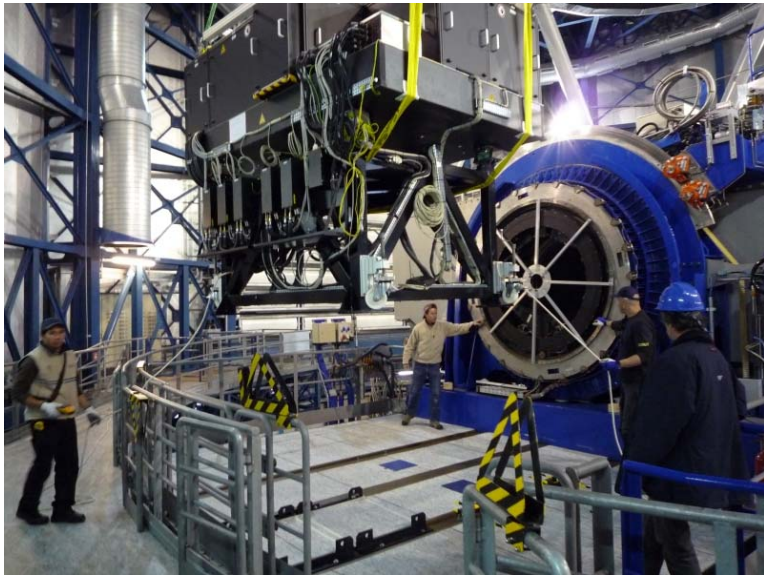
The challenge is to separate the variant from the common part and to manage the dependencies.

Example

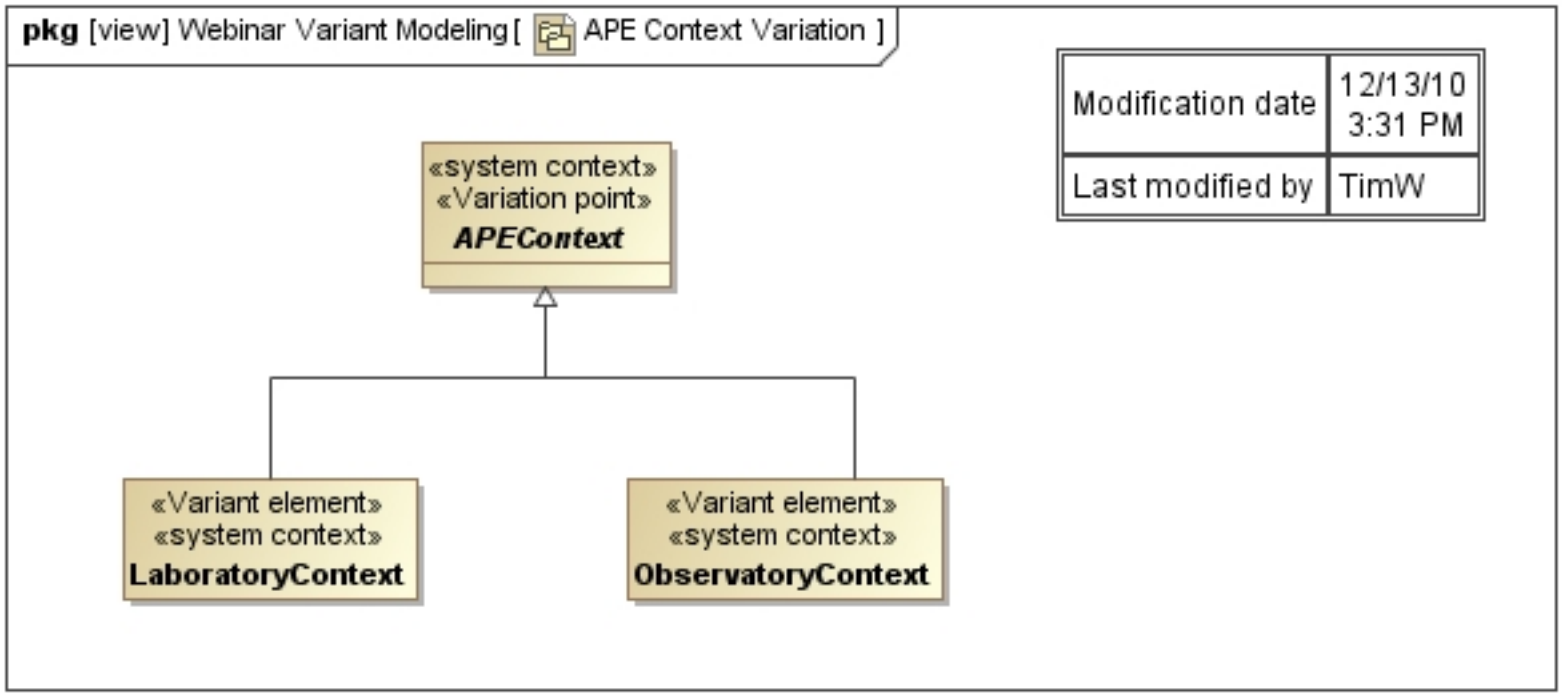
The APE system could be installed in the laboratory in Munich as well as directly at the VLT.

We use the concept of variant modeling to separate the different contexts in the model.





Simple model example



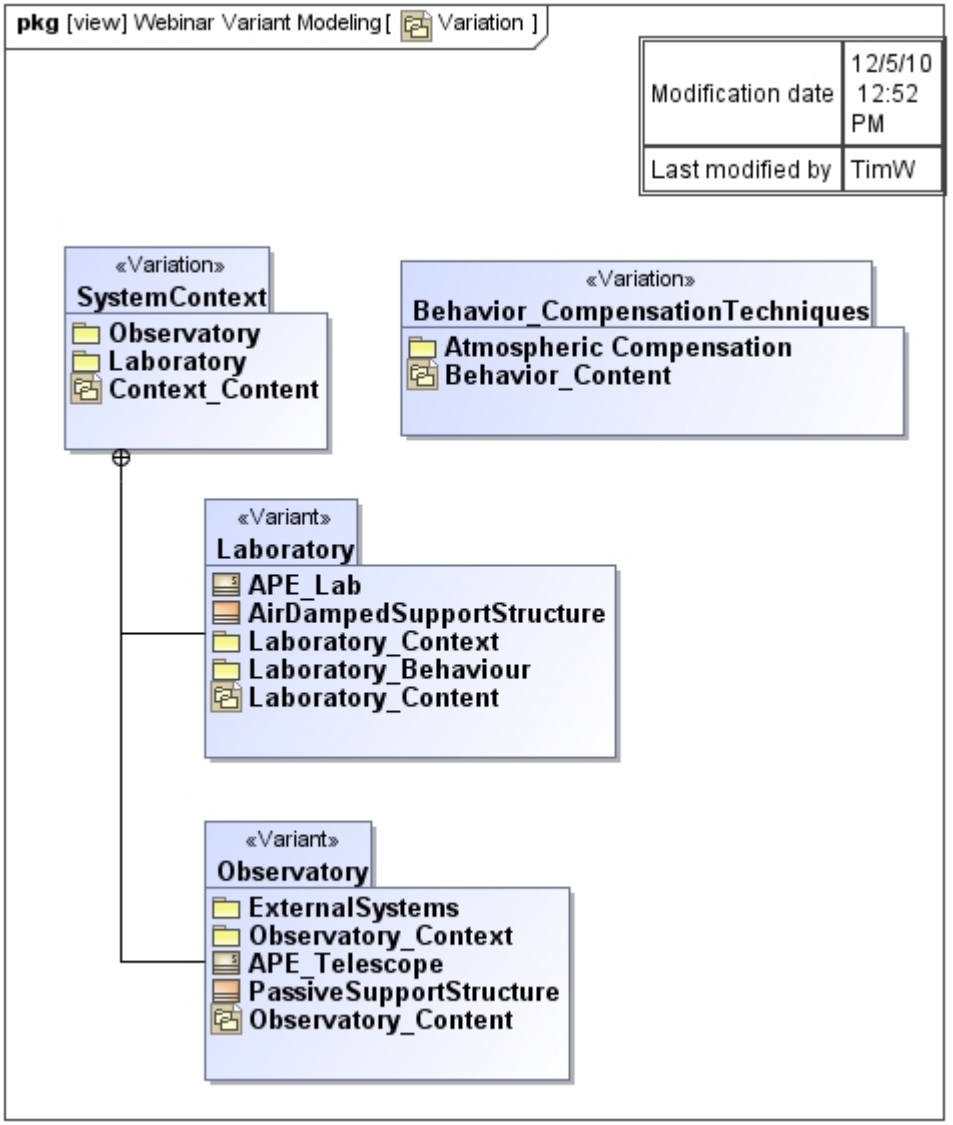
Agenda

- What is the SE^2 Challenge project about?
- Overview Variant Modeling
- **The SYSMOD Variant Profile for SysML**
- Variant Configurations
- Managing Complexity – Building simple views
- What's next?

Definitions

A **variation** contains a set of variants that have a common discriminator.

A **variant** is a complete set of variant elements that varies the system according to the variation discriminator.

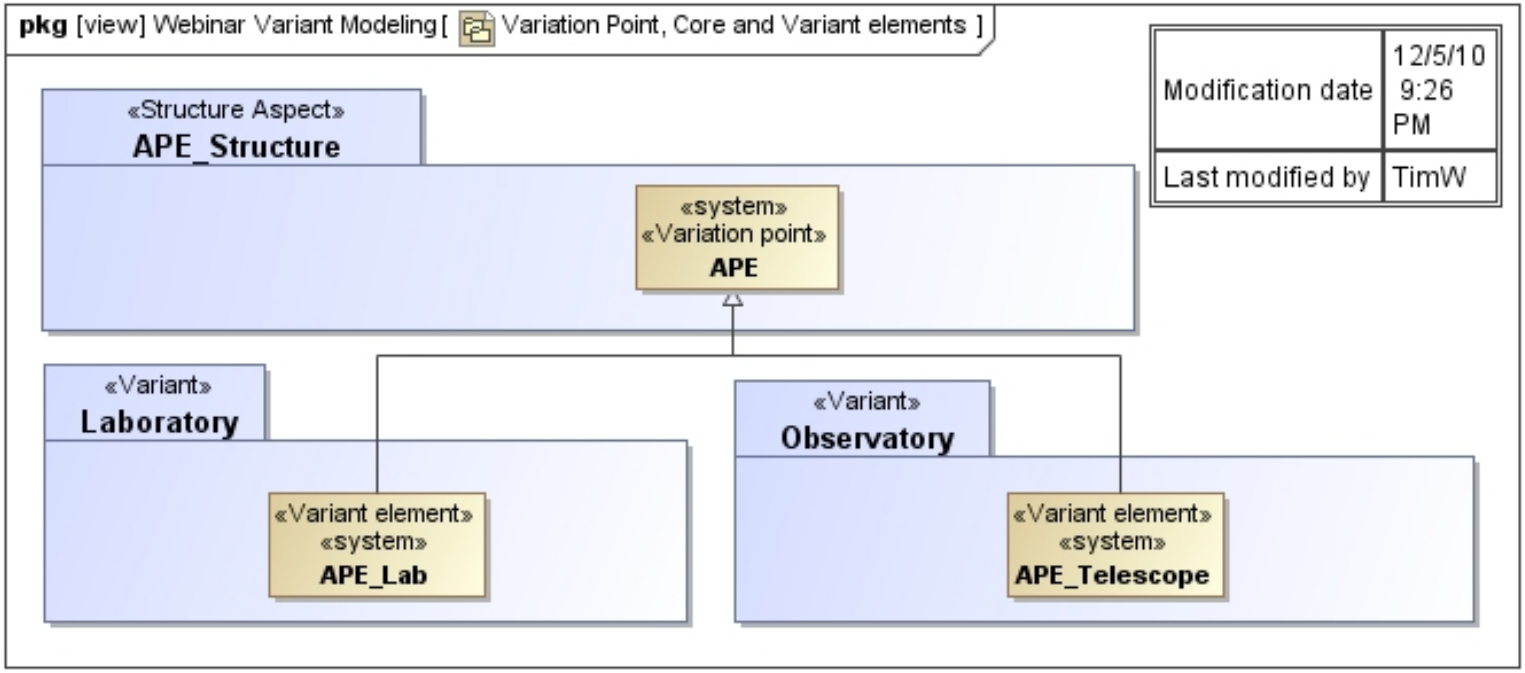


Definitions

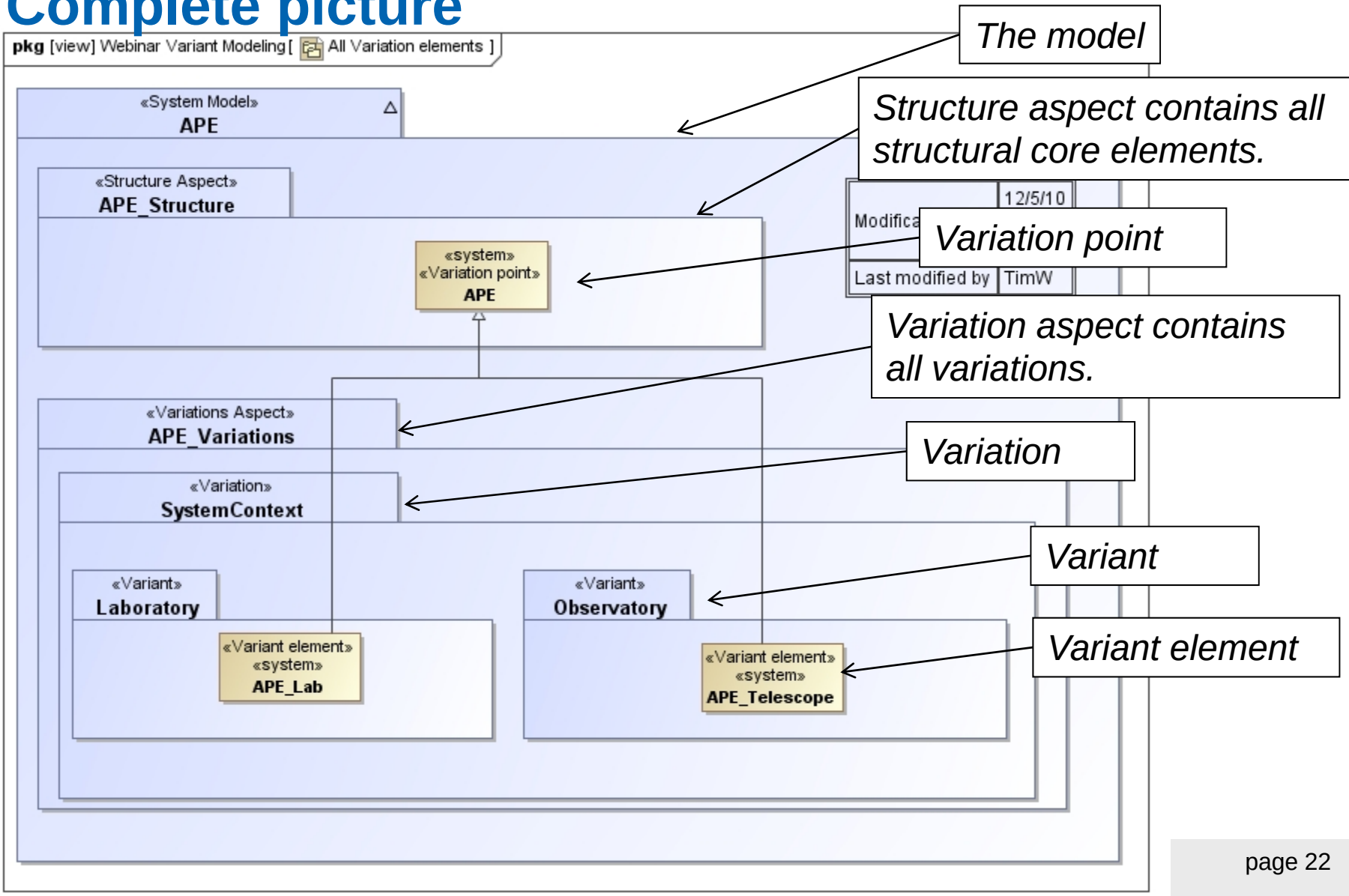
A **variation point** marks a core element as a docking point for a variant element.

A **variant element** is an element in a variant package.

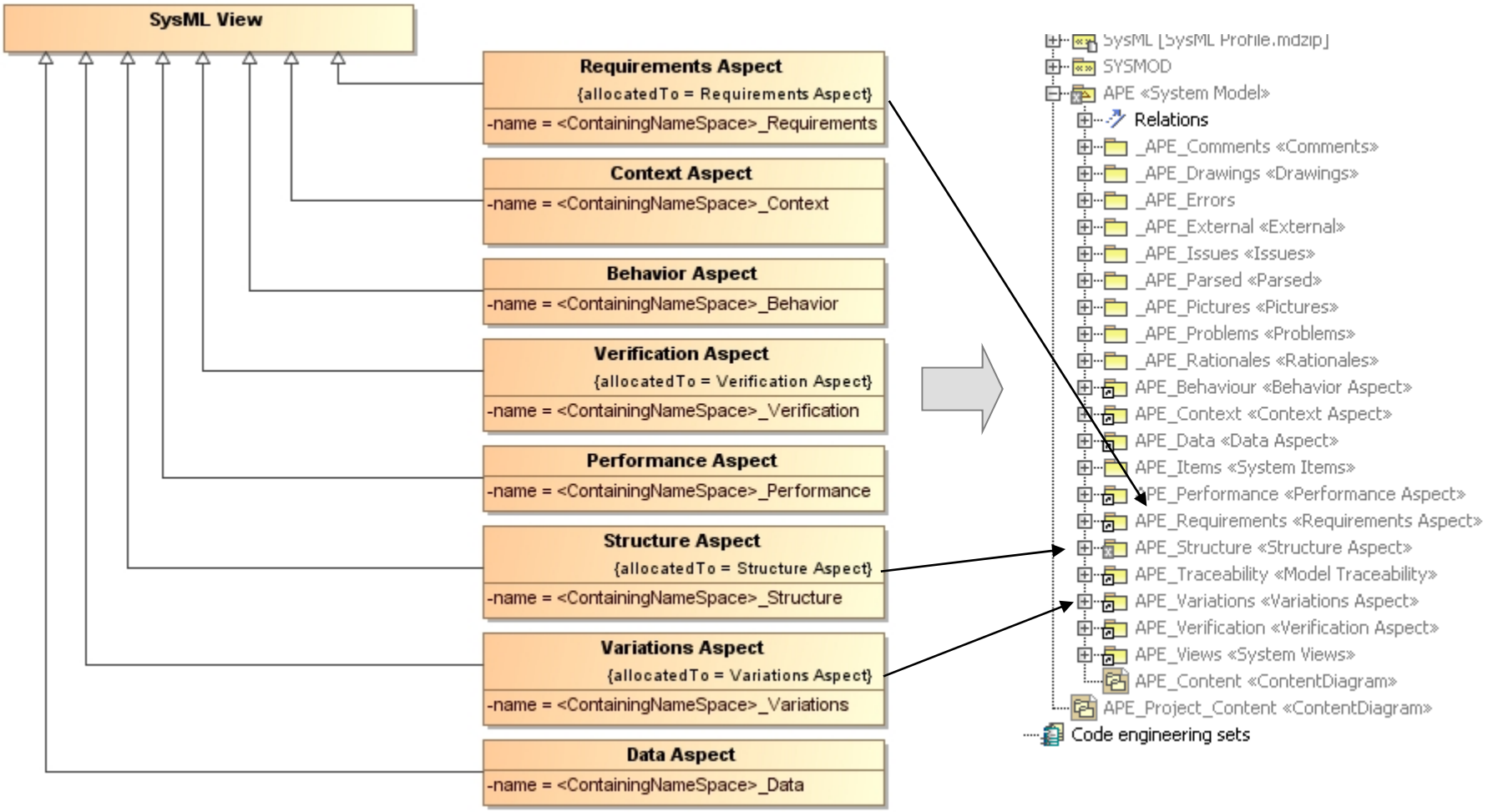
A **core element** is an element that is valid for all variants.



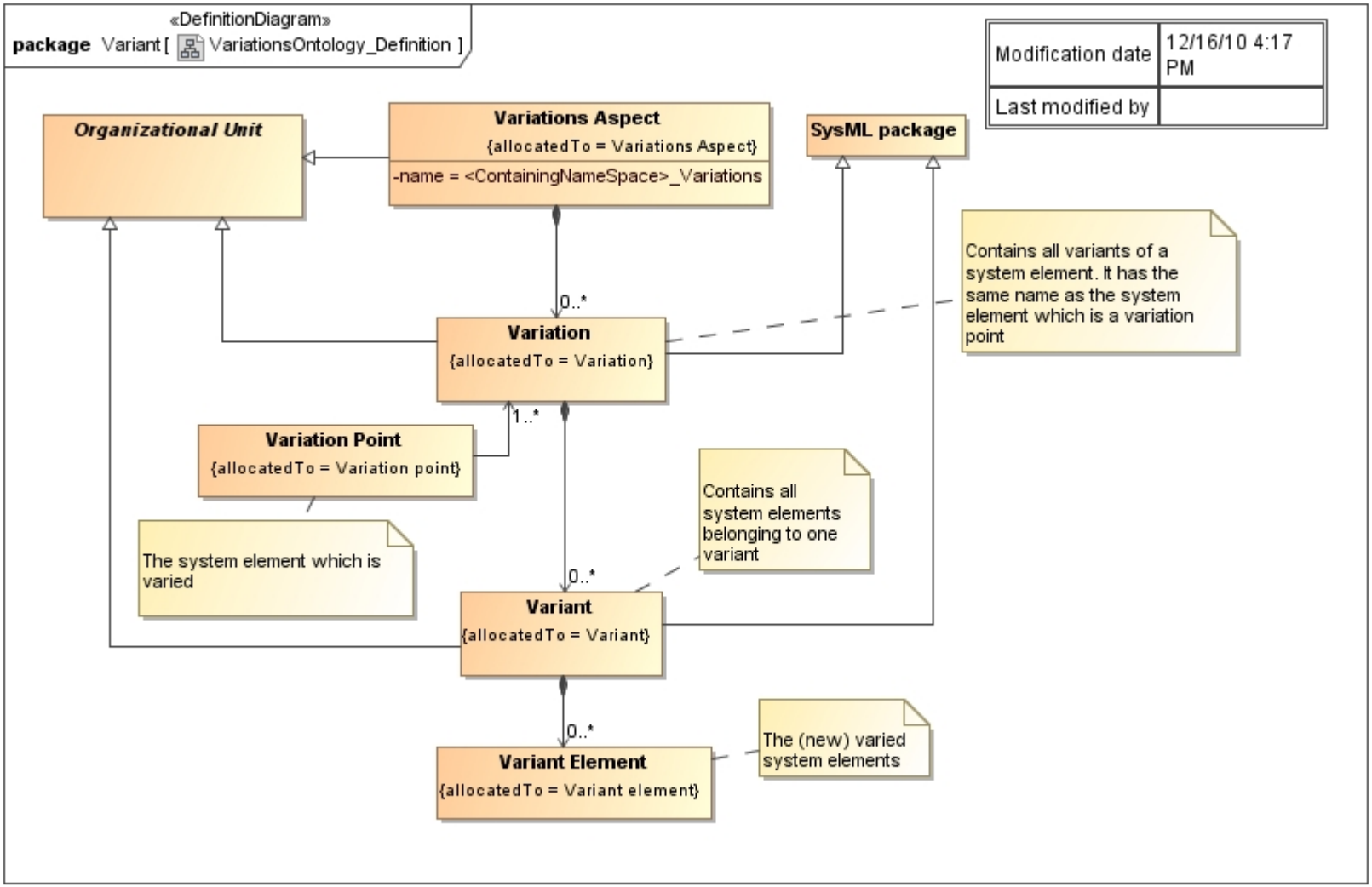
Complete picture



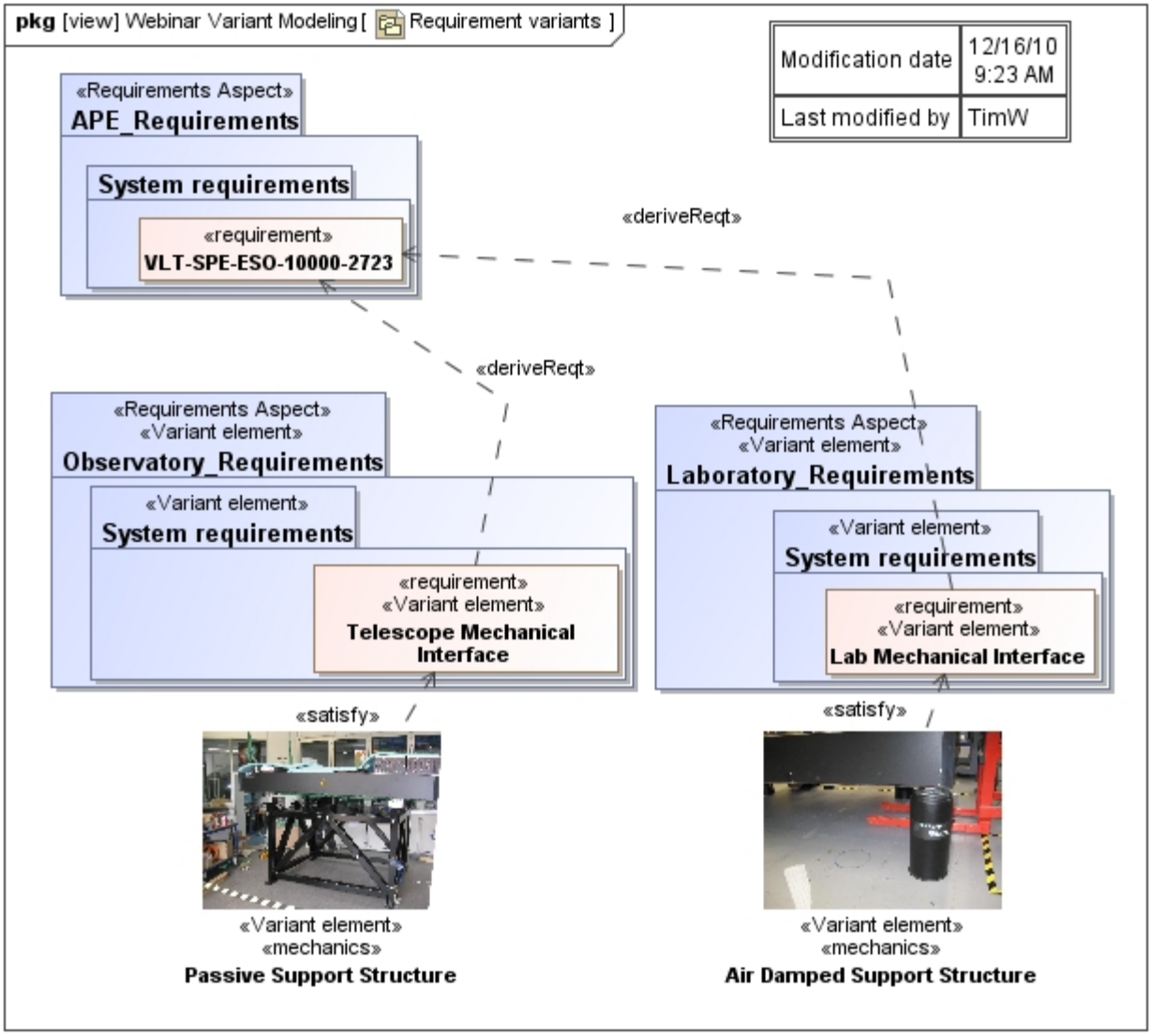
Ontology –
A formal definition of our modeling concepts



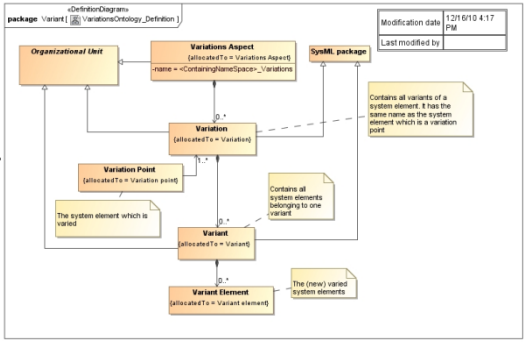
Variation Ontology

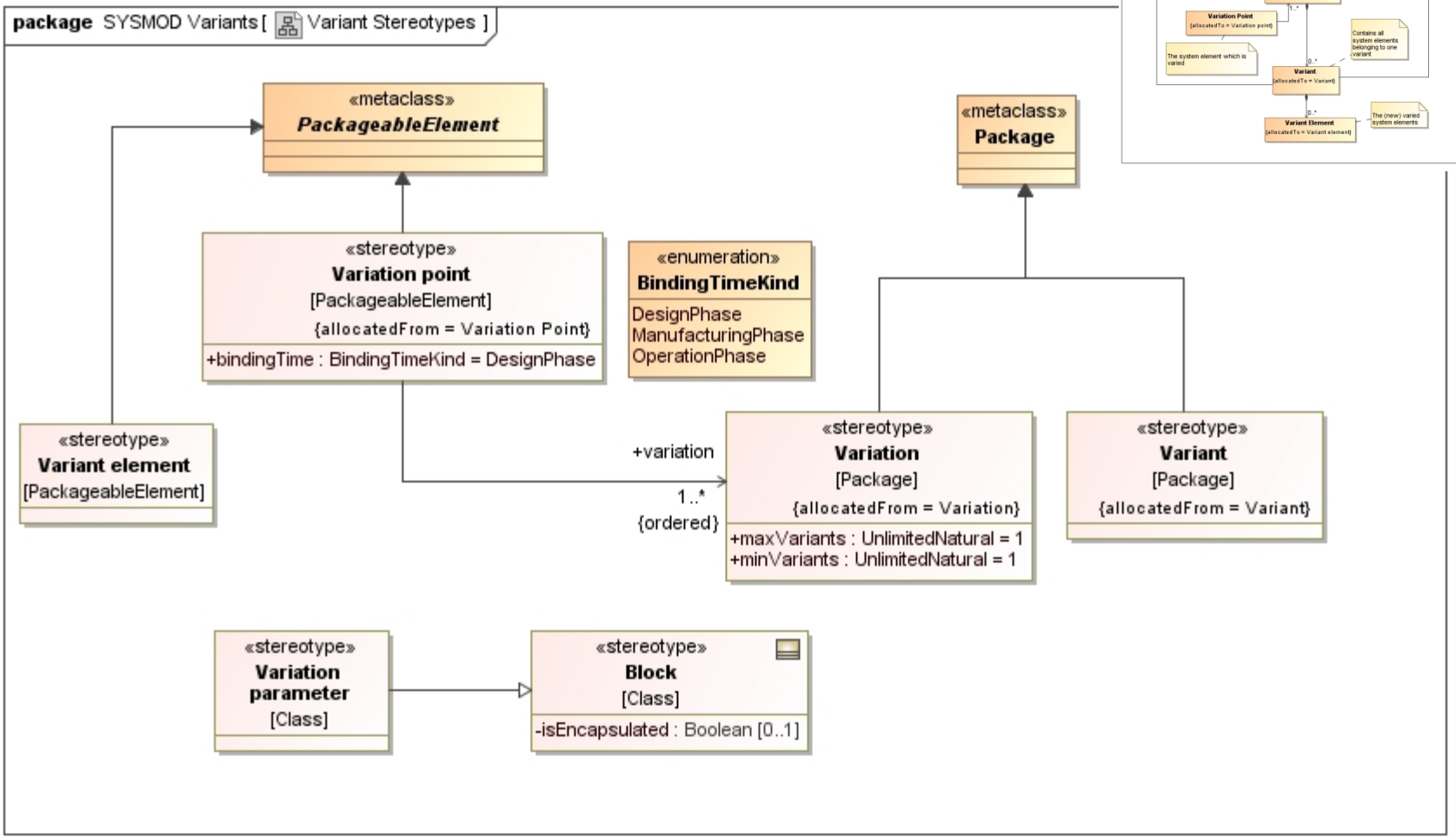


Example: Requirement variants

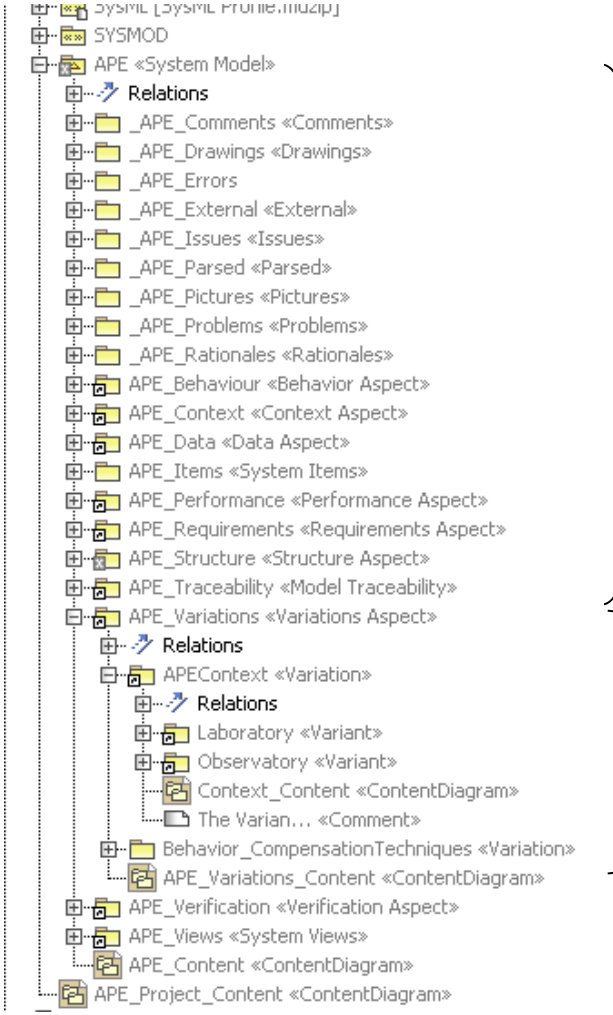


Profile for Variation Modeling





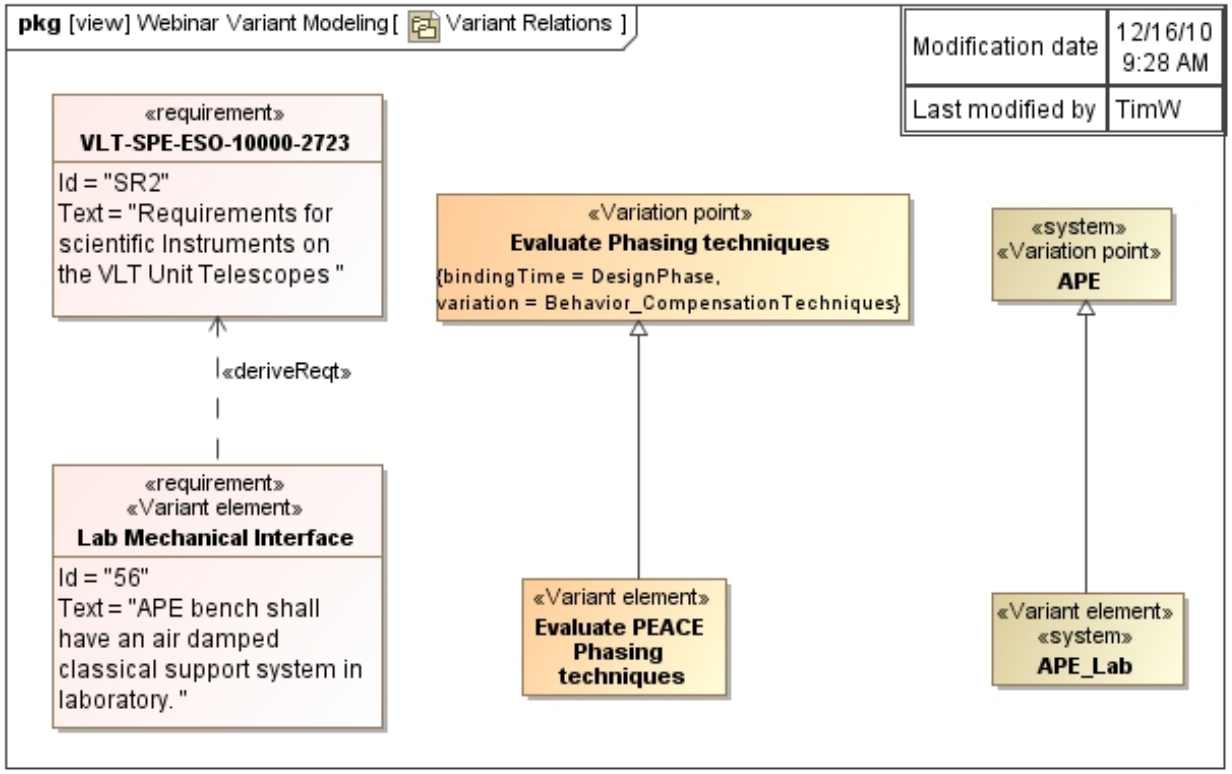
Separation of Concerns: Core and Variations



Core elements – valid for all variations

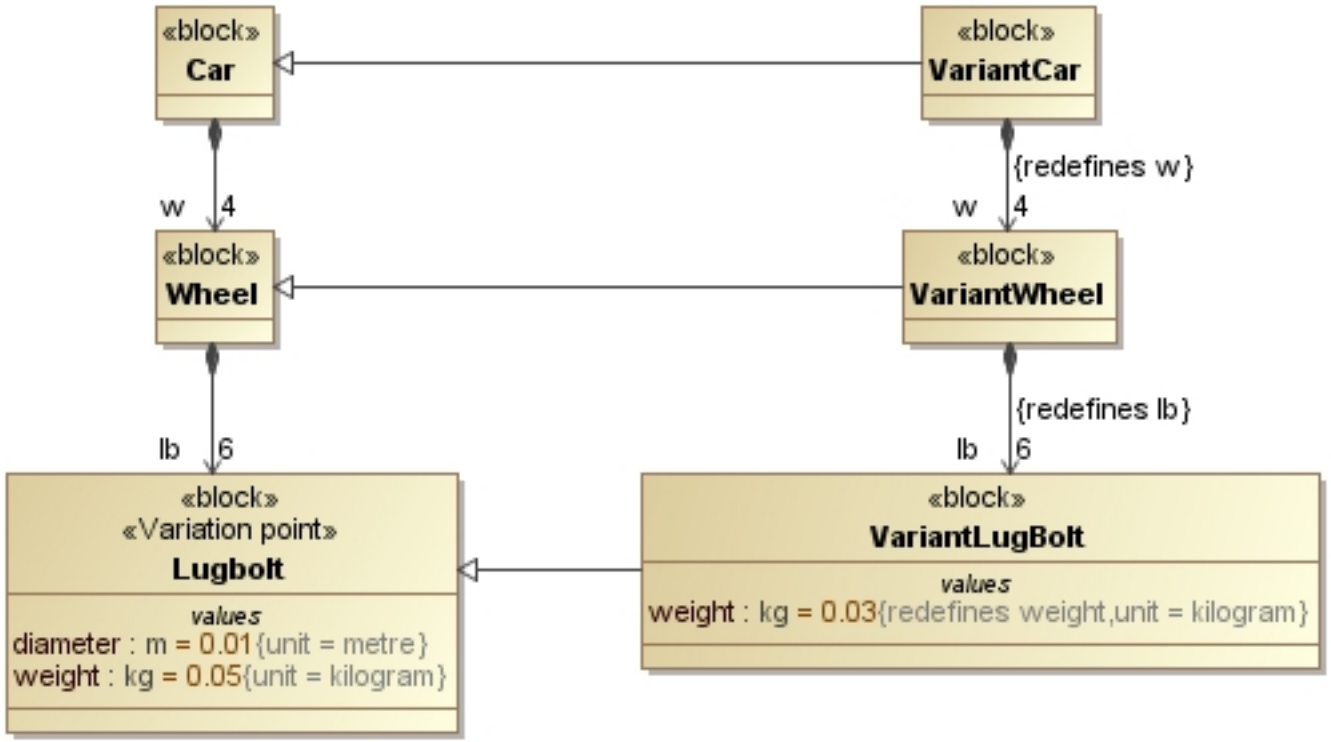
Variations – one root package for each variation and variant

Relations between variant and core elements

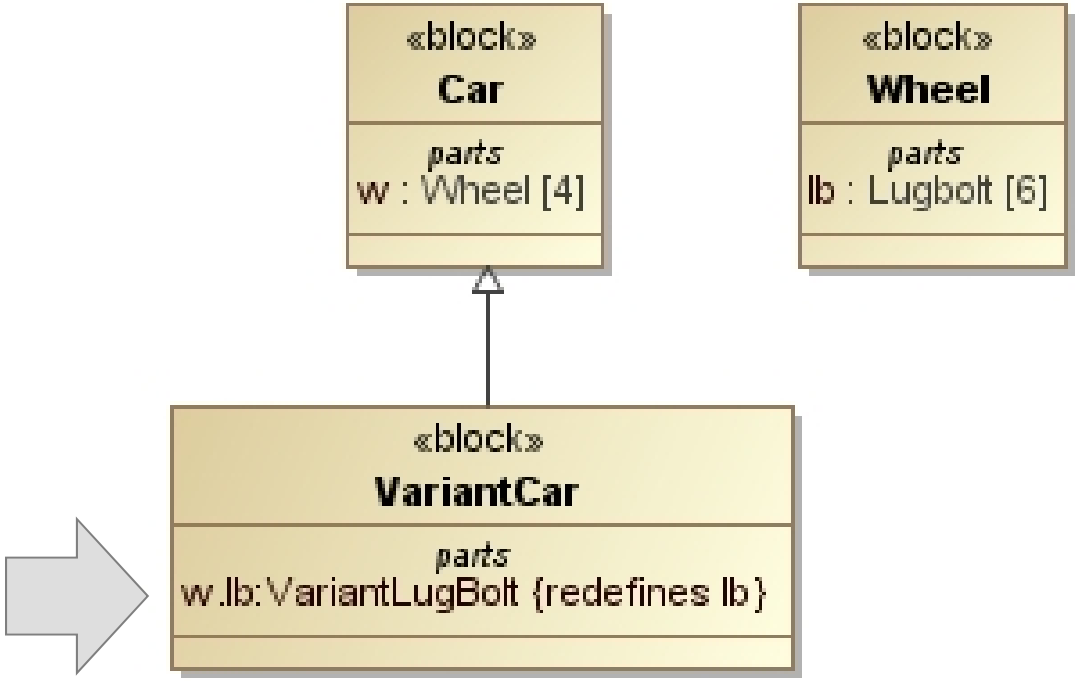


- No predefined guidelines for the relation type
- Degree of freedom of the relationtype depends on the application type of the model: automated analysis, model transformations, only human readable

Open issue – Modeling Generalizations



Proposal for a simplified Product Variation



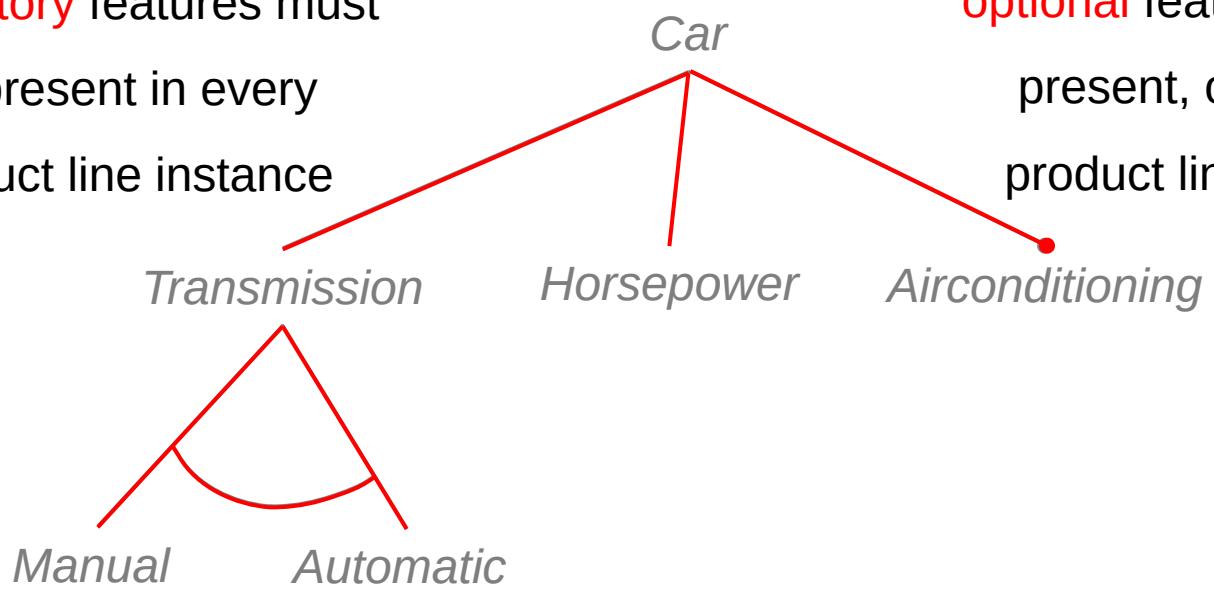
Agenda

- What is the SE^2 Challenge project about?
- Overview Variant Modeling
- The SYSMOD Variant Profile for SysML
- **Variant Configurations**
- Managing Complexity – Building simple views
- What's next?

FODA Feature Oriented Domain Analysis

mandatory features must be present in every product line instance

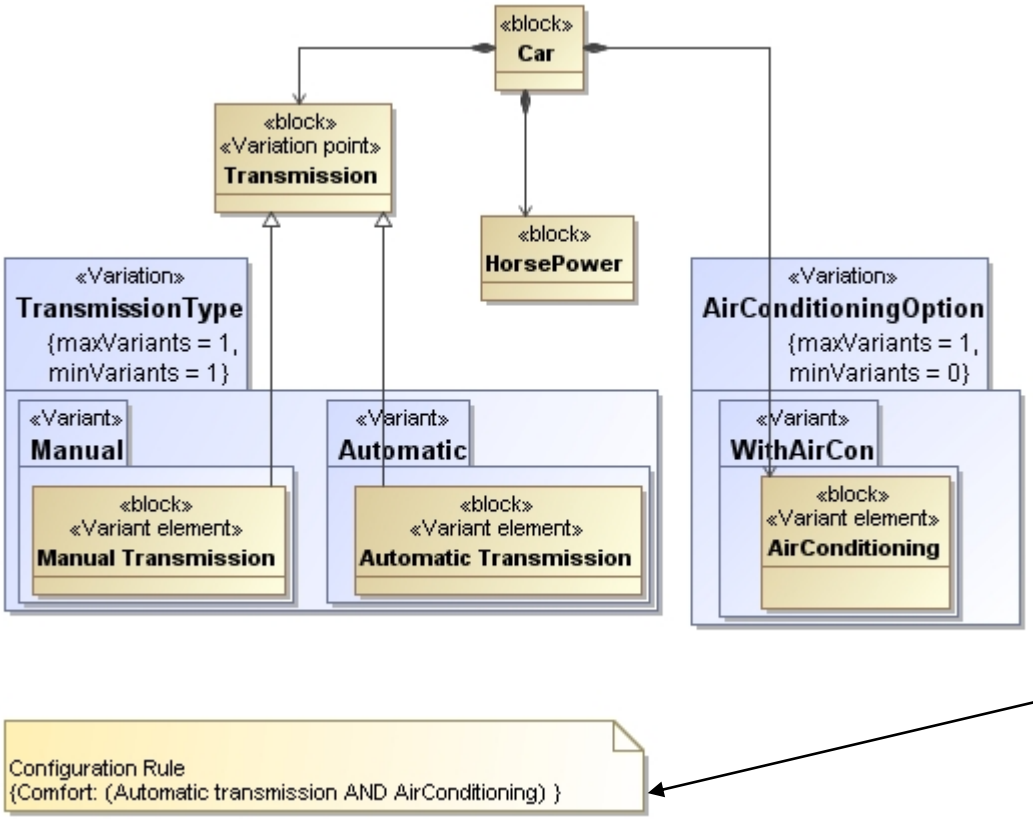
optional features may be present, or not, in a product line instance



alternative features define the scope for an exclusiveor choice of features

Source: Myra Cohen, Matthew Dwyer: Software Product Line Testing Part II : Variability Modeling

Example: Configuring variations



Rules formulated in a textual DSL and embedded in UML constraints ared defining the variation possibilities.

Agenda

- What is the SE^2 Challenge project about?
- Overview Variant Modeling
- The SYSMOD Variant Profile for SysML
- Variant Configurations
- **Managing Complexity – Building simple views**
- What's next?

Model2Model-Transformation (M2M)

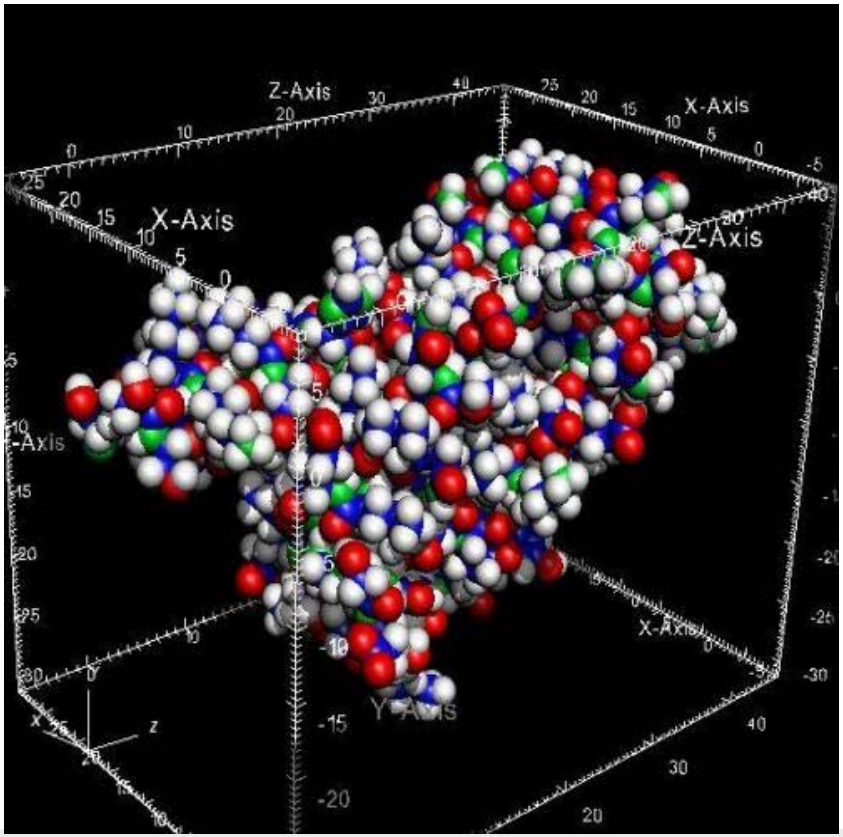
Even simple variations are resulting into complex configuration spaces. Necessary is a simple view for a selected configuration. This view could be produced by a M2M-Transformation.

3 variations are spanning a 3-dimen-sional configuration space and eventually many possible configurations.

A configuration is one point in the configuration space.

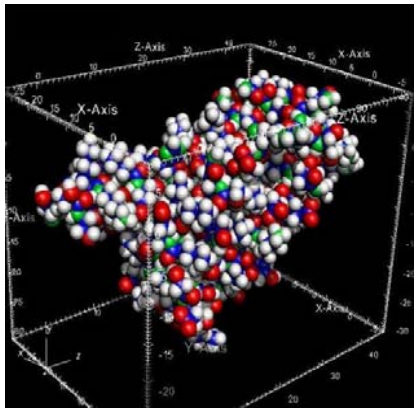
Aim of the model transformation:
Face-out of irrelevant details

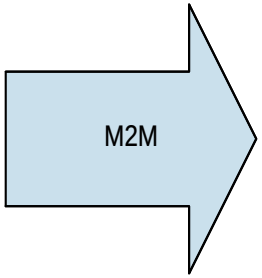
- Creation of a product model out of a product family model
- Elimination of non-existing variants and closure of variants because of superfluous abstractions

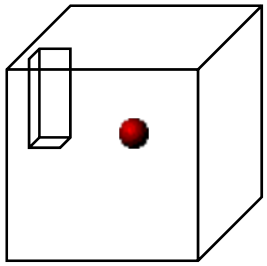


Categories of Model2Model-Transformations

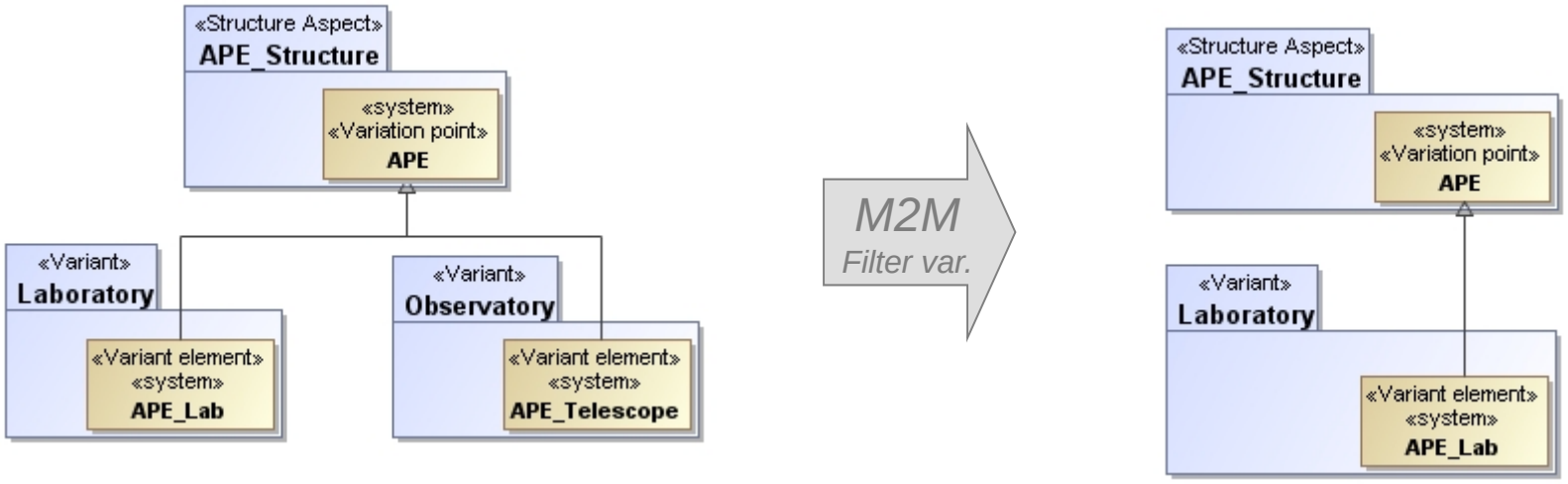
- View- vs. Copy-Strategy:
  - View: The transformation creates a view in the source model.
    -> Seperation of Product Line Engineering and Product Engineering
  - Copy: The transformation creates a new model
    -> Discard of variants during the development phase.
- Filter vs. Refactoring-Strategy
  - Filter: No more required model elements will be deleted (from the view or the copy)) by the transformation
    -> easy to apply, but some „ballast“ remains
  - Model-Refactoring: There exists not „the one and only“ transformation, but a set of adequate refactorings, with a corresponding non-trivial transformation
    -> best possible reduction, but hard to implement



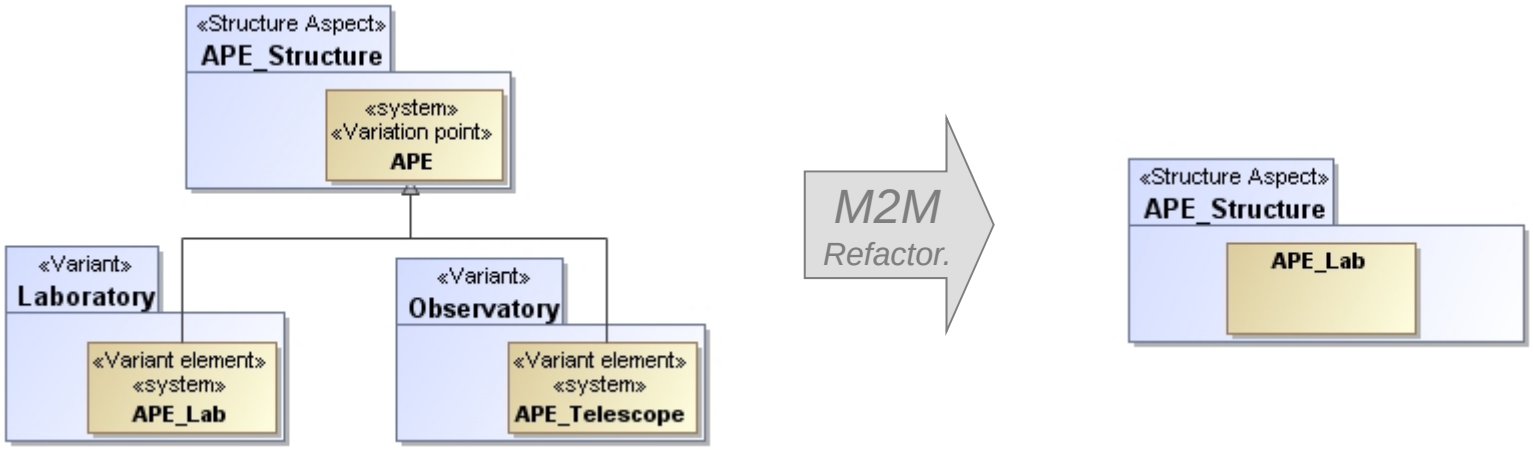




Example: Simple M2M approach



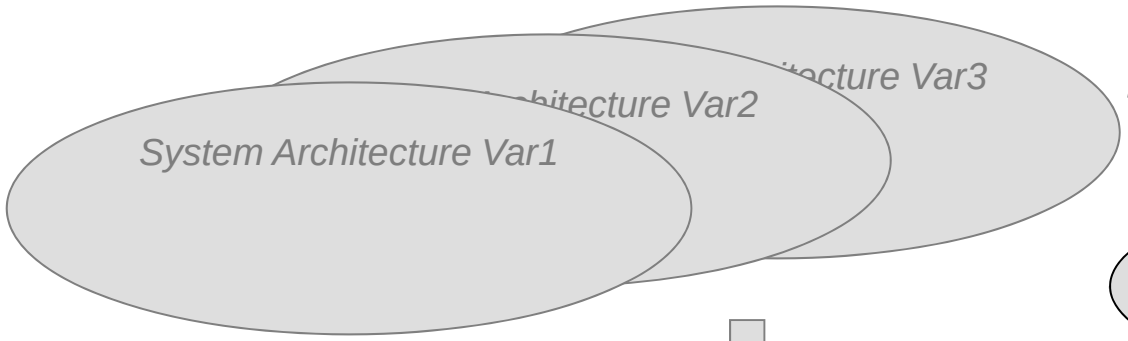
Example: M2M-Model-Refactoring approach

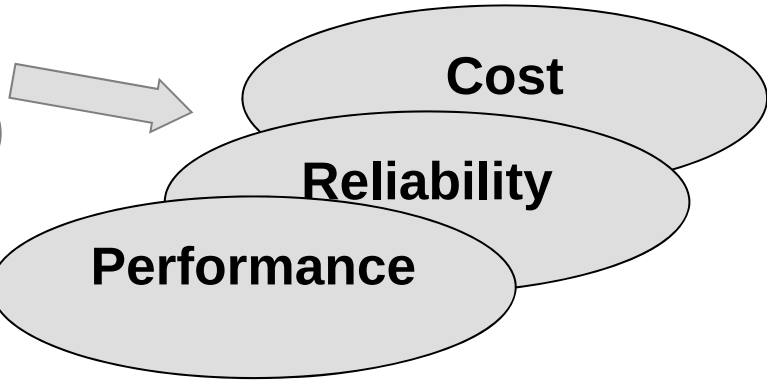


Open issue – Model2Model transformation

- Until now model transformations are only manually applied in the telescope model.
- Simple approach could be easily implemented:
  E.g. MagicDraw offers the Module concept. This could be used to hide all other variants and present only the elements belonging to that module/variant.
- Automatic model transformation by using transformation frameworks like OpenArchitectureWare, which is now part of the Eclipse Modelling Project
  http://www.eclipse.org/modeling/

Trade-Off Analysis for different alternatives

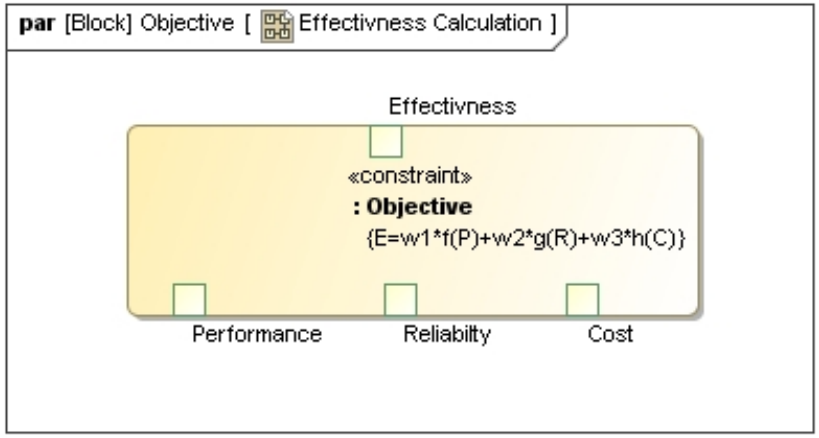




Feedback results

Components	*Var1*	*Var2*	*Var3*
Sensor	Sensor1	Sensor2	Sensor2
Actuator	Actuator1	Actuator2	Actuator2
Software	-	Standard	OpenSource

Criteria	*Weight*	*Var1*	*Var2*	*Var3*
Performance	0,3	5	10	20
Reliability	0,5	95%	98%	99%
Cost	0,2	10.000,00 €	20.000,00 €	20.000,00 €
Effectivness		**2001,975**	4003,49	4006,495



Source: Sanford Friedenthal:, ***Advancing Systems Engineering Practice Using Model Based Systems Development***

Agenda

- What is the SE^2 Challenge project about?
- Overview Variant Modeling
- The SYSMOD Variant Profile for SysML
- Variant Configurations
- Managing Complexity – Building simple views
- **What's next?**

What's next?

- Generalization issue ➔ Fix by SysML RTF
- M2M transformation ➔ Find resources to develop transformation
- Trade-Off Analysis ➔ Make a Trade-Off analysis in the Telescope context using the new simulation toolkit from MagicDraw

Live Demo of the Model

- Please fasten seatbelts - setting up the system…