TMT
Thirty Meter Telescope

NASA

A Model-Based Approach to Error Budgeting using SysML

Using OMG SysML for performing error accounting for complex systems

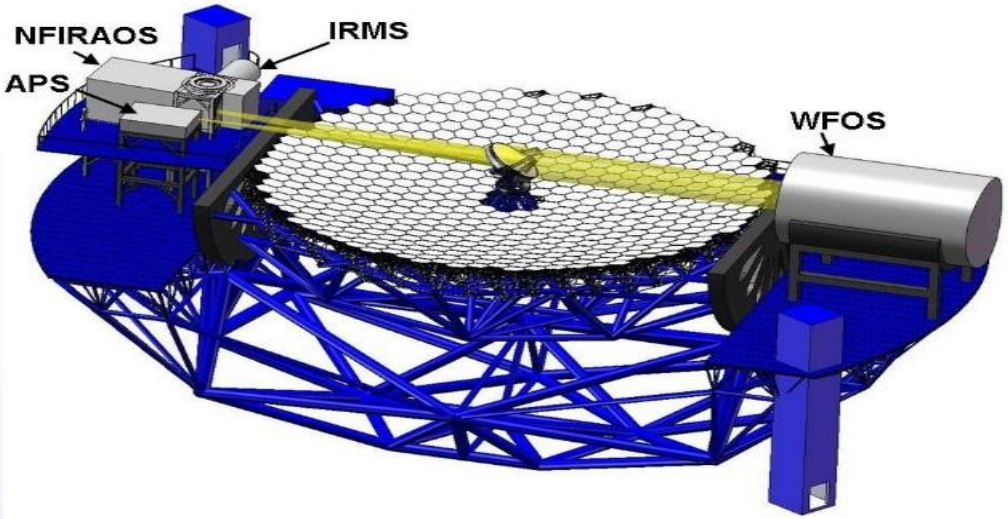
Sebastian Herzig, Frank Dekens, Robert Karban, Mitchell Troy

Outline

- Context
- What is Error Budgeting?
- Motivating Example
- Error Budgeting in SysML
  - Error Breakdown Tree
  - Error Roll-Up Pattern
  - Tying to Requirements & Product Breakdown Structure
- MagicDraw Customizations
- Summary & Conclusions

Context

- Alignment and Phasing System (APS)
  - Sensor responsible for measuring the pre-adaptive optics wavefront quality
  - APS (and AO) team uses MBSE with SysML to analyze requirements, produce design, and perform analysis



Error Budgeting & Error Analysis

- Track how accurate expected system performance will be compared to the requirements

- Error budgeting:
  - Flow down requirements and allocations
  - Roll-up capabilies or Current Best Estimates (CBE)
  - Track margins and reserve
    - Examples: Mass, Power, Throughput, timing, Pointing Alignment, measurement errors, performance

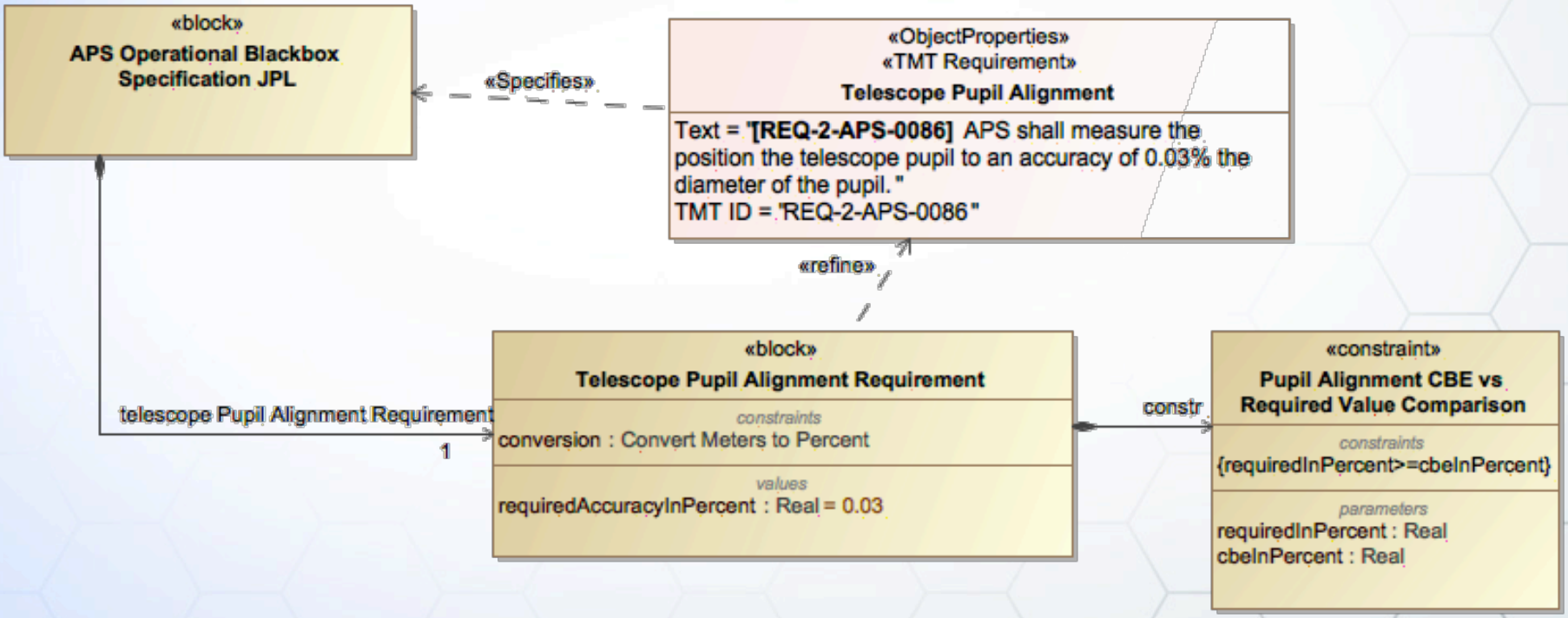
Challenges

- Error budget analysis involves multiple, related artifacts:
  - Requirements ← required accuracy
  - System design ← current best estimates, parameters
  - Error decomposition ← roll-up definition

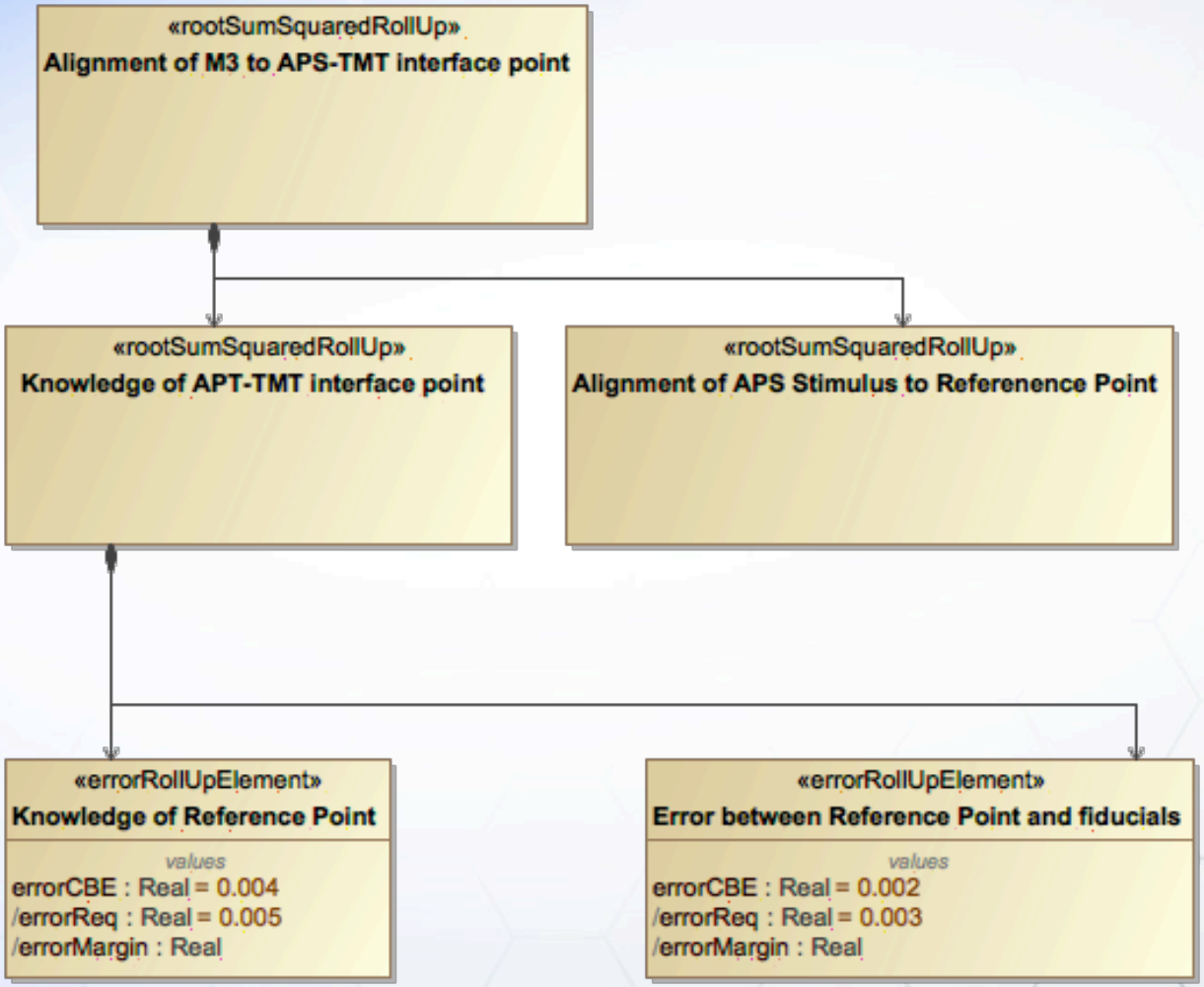
- CBEs of errors *characterize* errors on properties of the system
- Decomposition of error != product breakdown structure
  ➔ Related, but different hierarchies

Motivating Example

- M3 to APS-TMT interface pupil alignment error: "*APS shall measure the position of the telescope pupil to an accuracy of 0.03% of the diameter of the pupil.*"

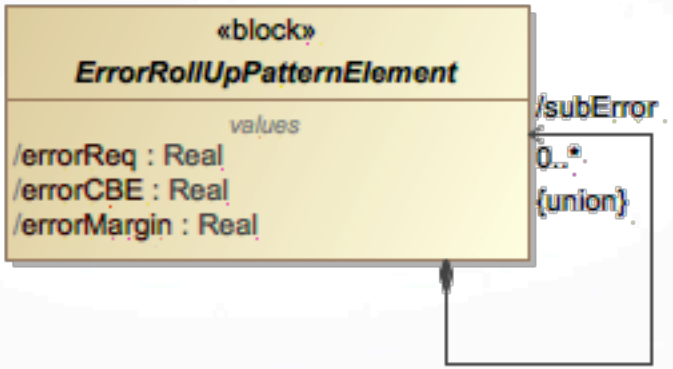


Error Budgeting in SysML: Error Breakdown Tree (Excerpt)



Error Roll-Up Pattern

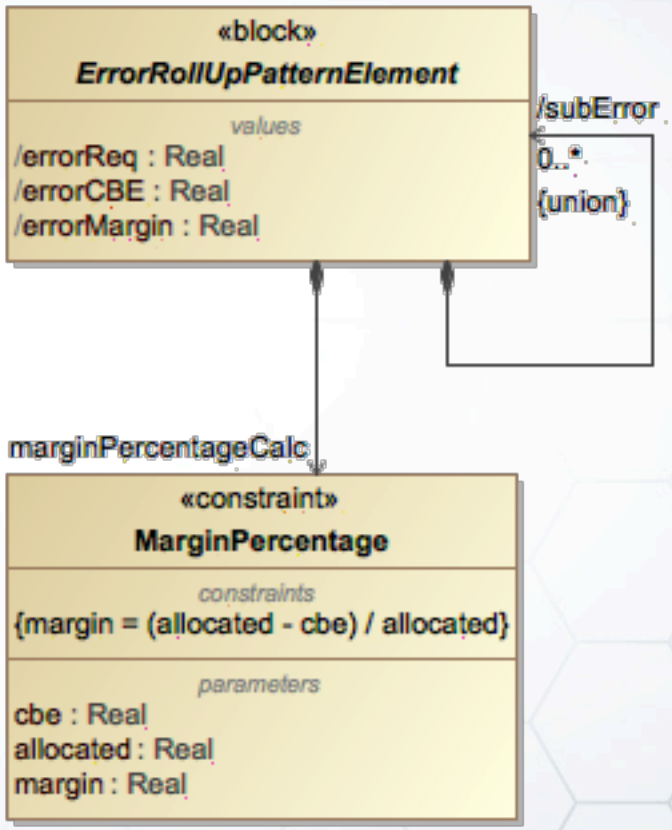
- Similar to mass roll-up pattern



Margin Calculation

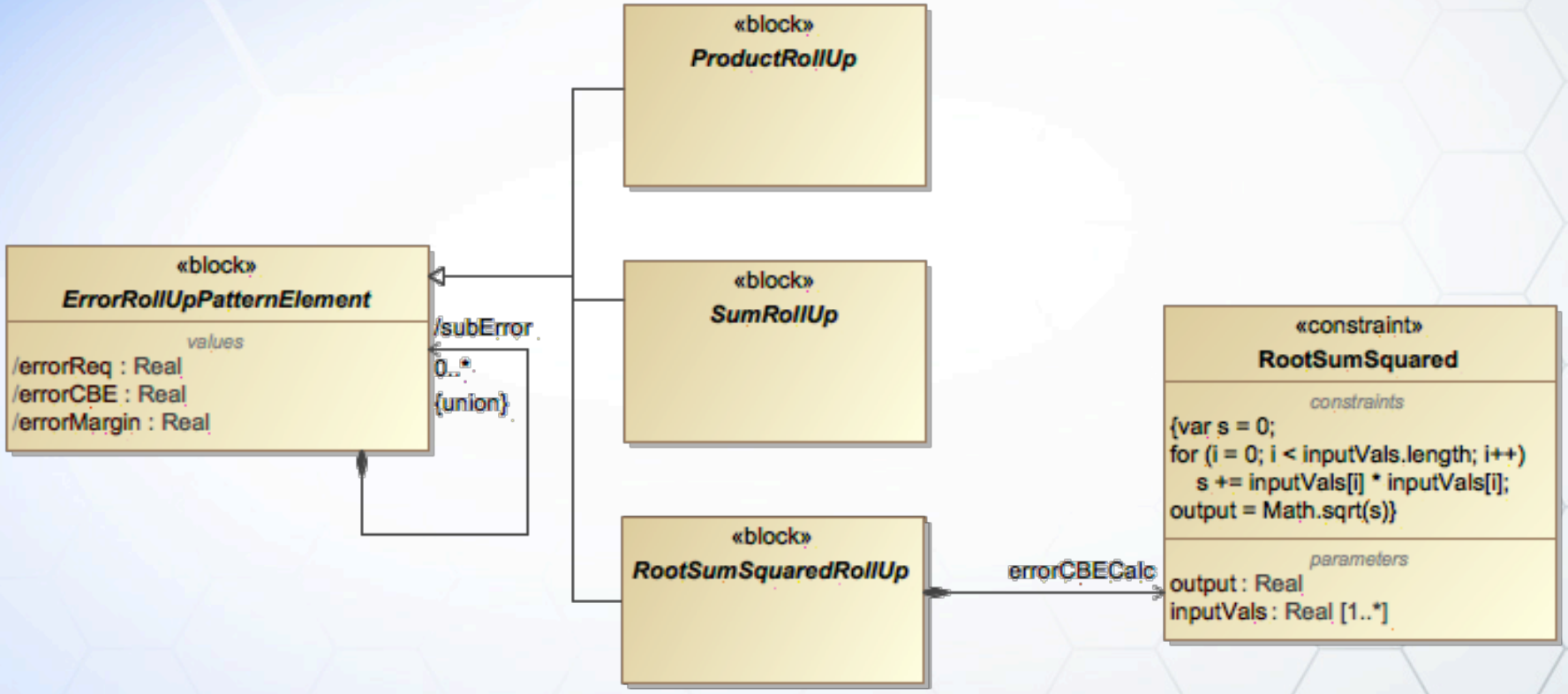
- Margin depends only on CBE and and allocation

$$margin = allocated - cbe / allocated$$



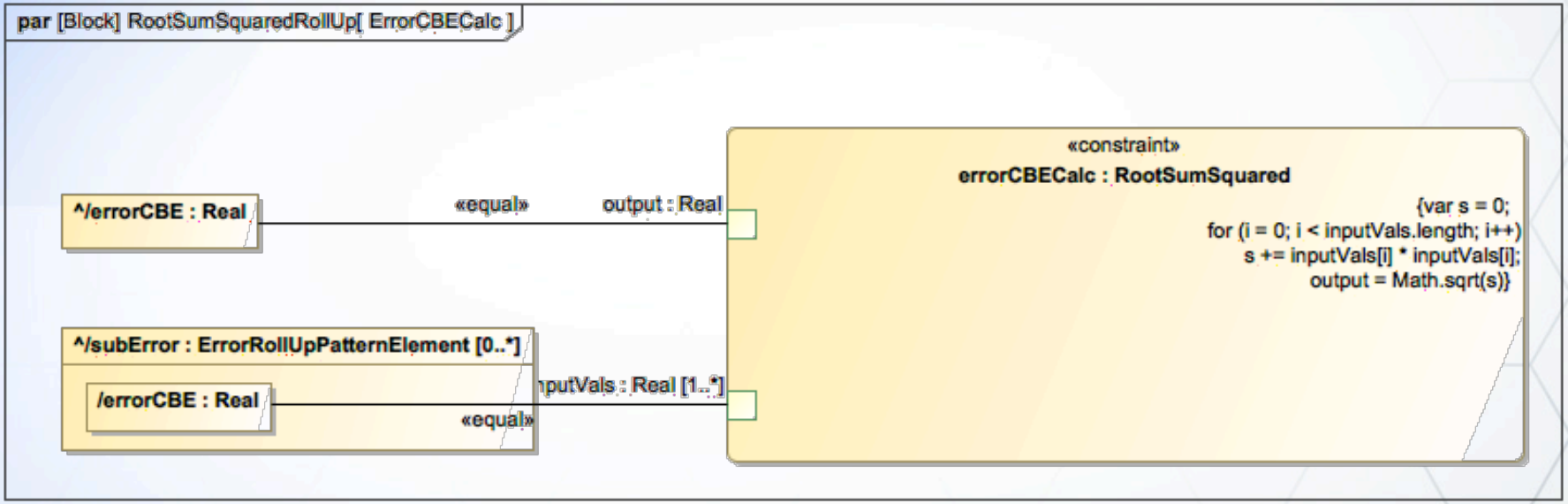
Error Roll-Up Types

- Different roll-up calculations depending on context

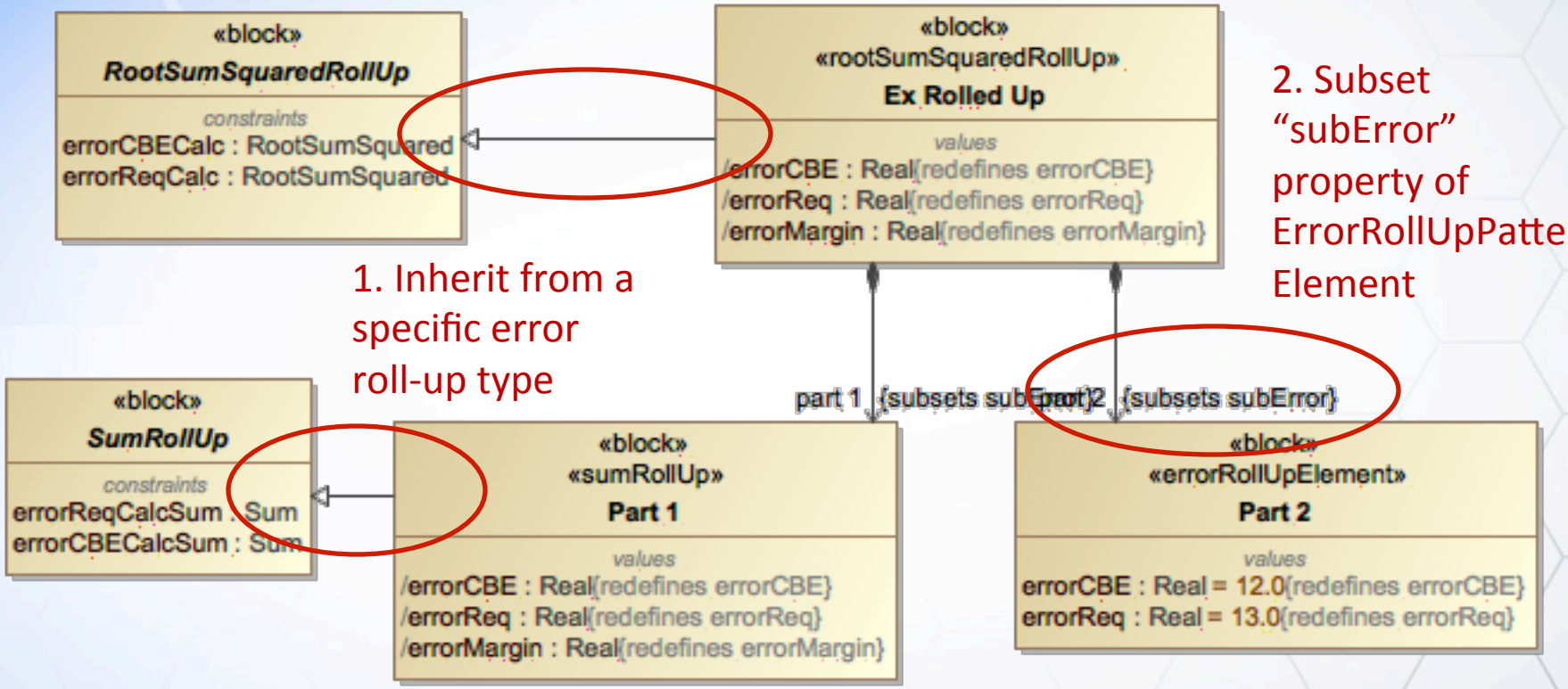


Error Roll-Up Types

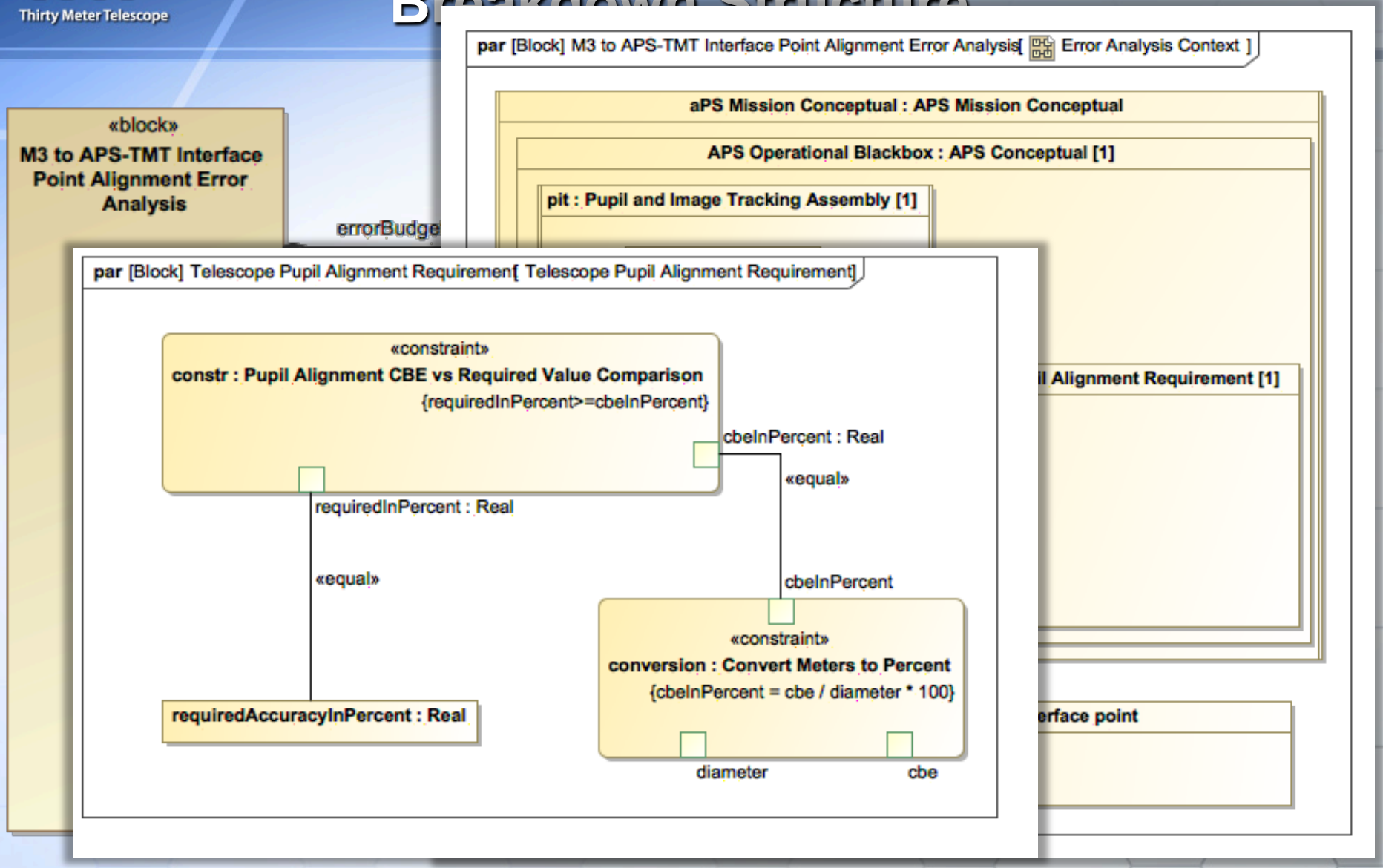
- Different roll-up calculations depending on context



Pattern Application

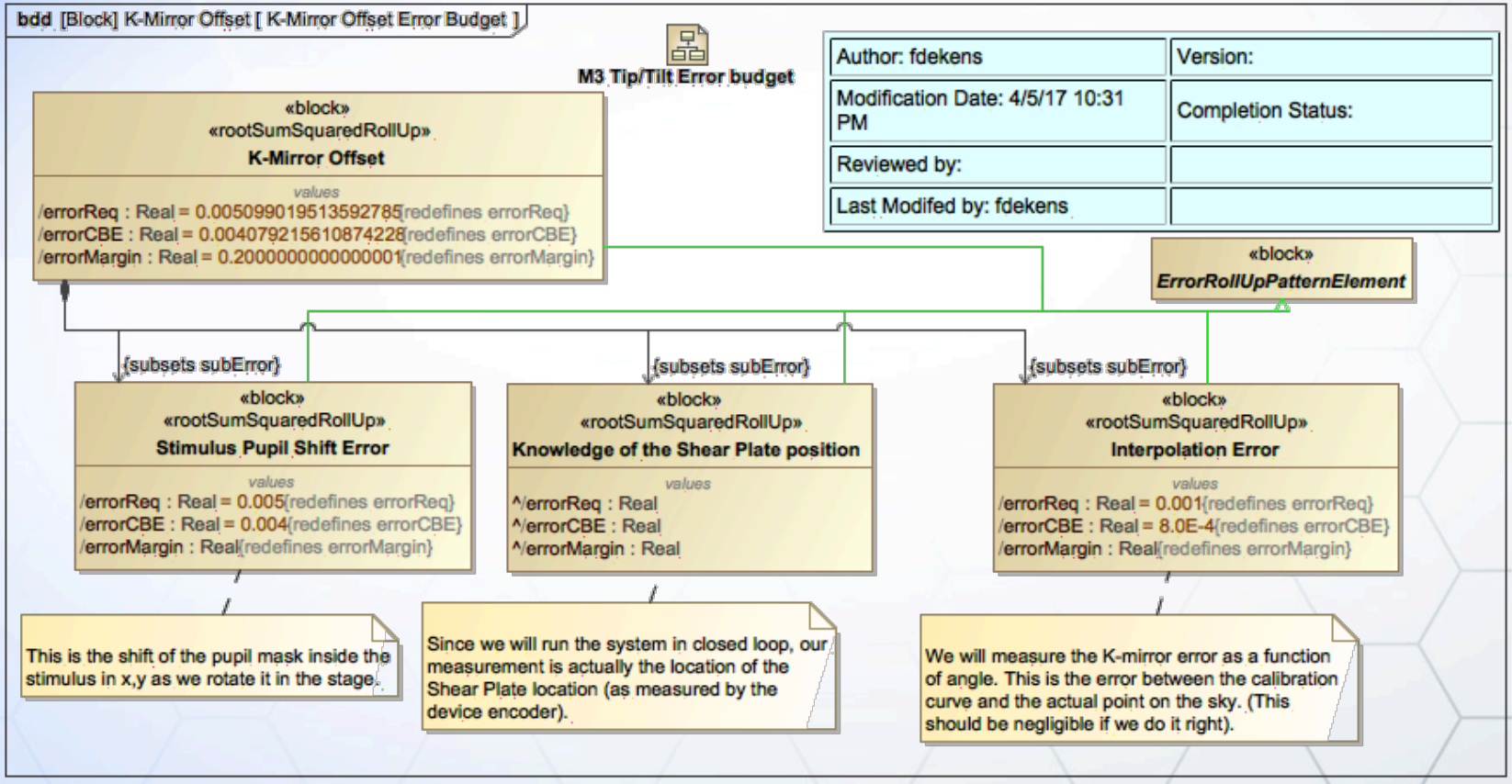


Tying to Requirements & Product Breakdown Structure



Solving & Results

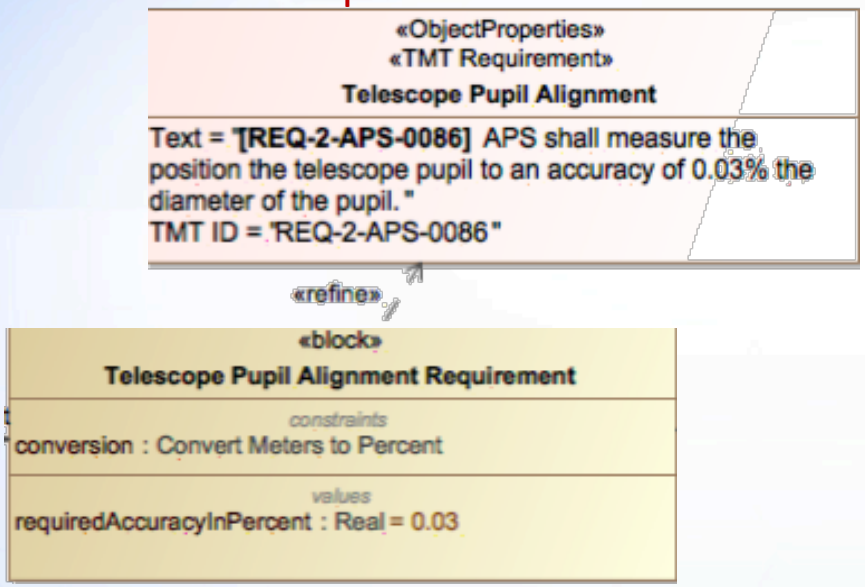
- Parametric solver used for solving set of equations specified in model – here, *Cameo Simulation Toolkit* was used



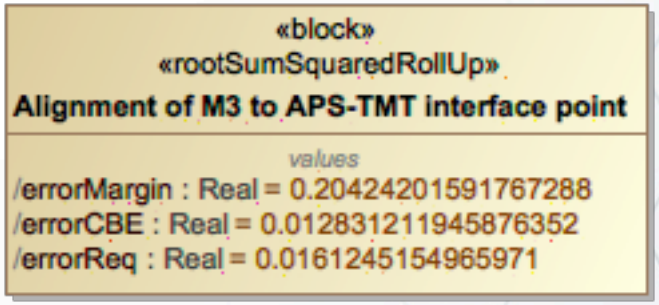
Solving & Results

- Parametric solver used for solving set of equations specified in model – here, *Cameo Simulation Toolkit* was used

1. Formalized requirement



2. Automated roll-up

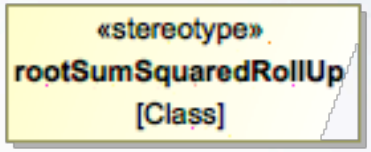


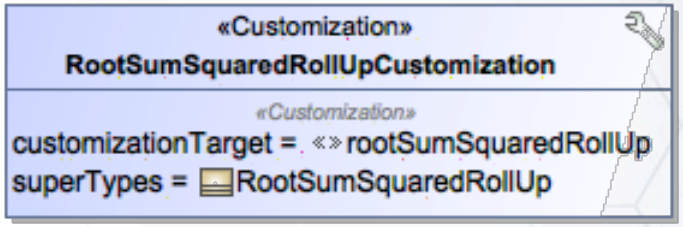
3. Automated verification of requirement

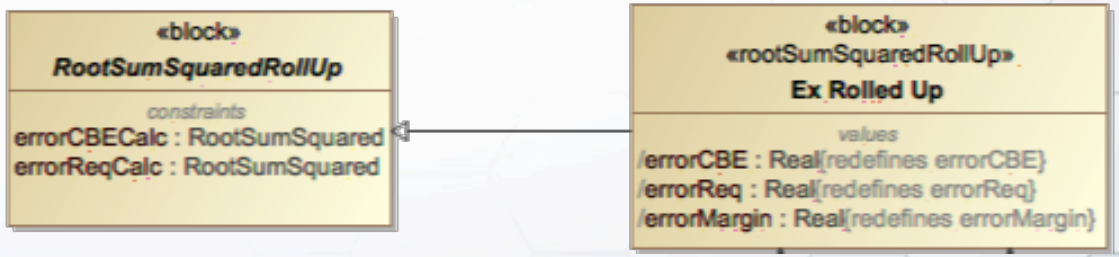
#	Name	Constr : Pupil Alignment CBE Vs Required Value Comparison
1	m3 to APS-TMT Interface Point	fail

MagicDraw Customizations

- If we hide inheritance, how do we know which type of error roll-up was applied?
- Also, inheritance cumbersome
- Used stereotypes and MagicDraw DSL customizations to solve both
- Automatic inheritance from a roll-up type when stereotype is applied
  - Unfortunately, no solution for subsetting…







How Does the Excel Method Compare to SysML?

- Typical Excel Error Budget
  - Pros:
    - Every one knows Excel!
    - Easy to make a copy for what-ifs
    - Easy to make changes
    - Easy to make yet an other error budget (copy one from an other project, or from online)
    - Extensive math libraries
    - Existing tools (queries, linking, etc.)
- Proposed SysML Error Budget
  - Pros:
    - Tied to requirements
    - Tied to system design
    - Tied or system parameters
    - Collaborative
    - Version controlled
    - Standardized through patterns
    - Can be extended to math languages (like Matlab, etc.)

Summary & Conclusions

- Demonstrated method and pattern for performing error budget analysis within a SysML model
  - Capture decomposition of error
  - Perform roll-up of current best estimates of leaf nodes
  - Calculate margins & verify requirements

- Method formally ties together requirements on accuracy, system design and relations between errors in an error budget
  - Safer → integration provides less danger of inconsistencies
  - Comes at a cost → careful modeling required

Acknowledgments

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration and NoMagic.

The TMT Project gratefully acknowledges the support of the TMT collaborating institutions. They are the Association of Canadian Universities for Research in Astronomy (ACURA), the California Institute of Technology, the University of California, the National Astronomical Observatory of Japan, the National Astronomical Observatories of China and their consortium partners, and the Department of Science and Technology of India and their supported institutes. This work was supported as well by the Gordon and Betty Moore Foundation, the Canada Foundation for Innovation, the Ontario Ministry of Research and Innovation, the National Research Council of Canada, the Natural Sciences and Engineering Research Council of Canada, the British Columbia Knowledge Development Fund, the Association of Universities for Research in Astronomy (AURA) and the U.S. National Science Foundation.

References

- Open Source TMT model: https://github.com/Open-MBEE/TMT-SysML-Model
- A Practical Guide to SysML, 3rd Edition, Chapter 17 by Friedenthal, Moore, and Steiner

Backup