

Planetary Utilities

INCOSE October SysML v2 Exchange

AI-Assisted, Selective Model Migration & Starforge Academy

Dr. Johannes Gross, CEO



Johannes Gross

Founder CEO

- Worlds first automated Spacecraft design
- 7+ years at NASA
- Flight hardware on Mars and ISS



Robert Karban

CTO

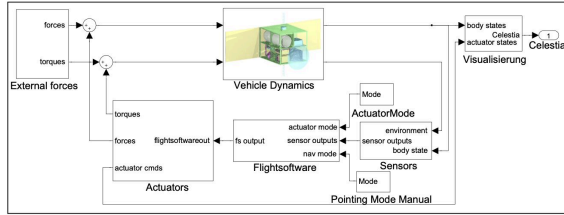
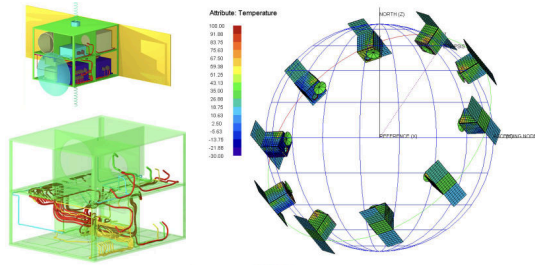
- Decades of Industry Experience
- Delivered large control systems
- Renowned Expert MBSE



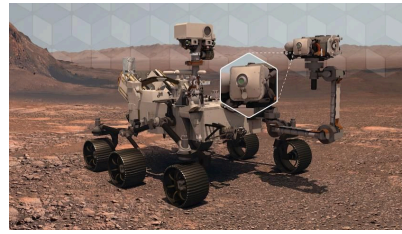
Two experts that spent their careers on this problem.



Universität
Stuttgart



- Group lead for Software Development
- PhD Thesis on Satellite Design Languages

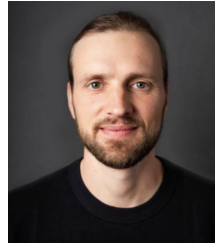
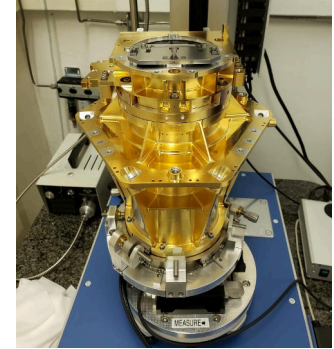


JPL

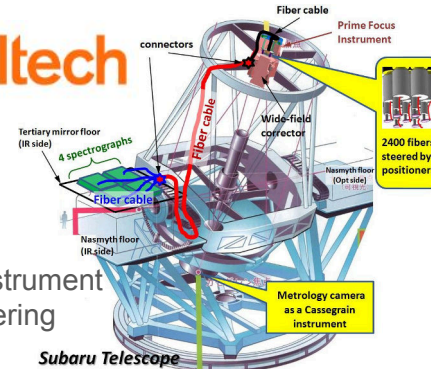
Jet Propulsion Laboratory
California Institute of Technology

PIXL Mars Rover Instrument I&T

Johannes



Caltech



PFS Instrument
Engineering

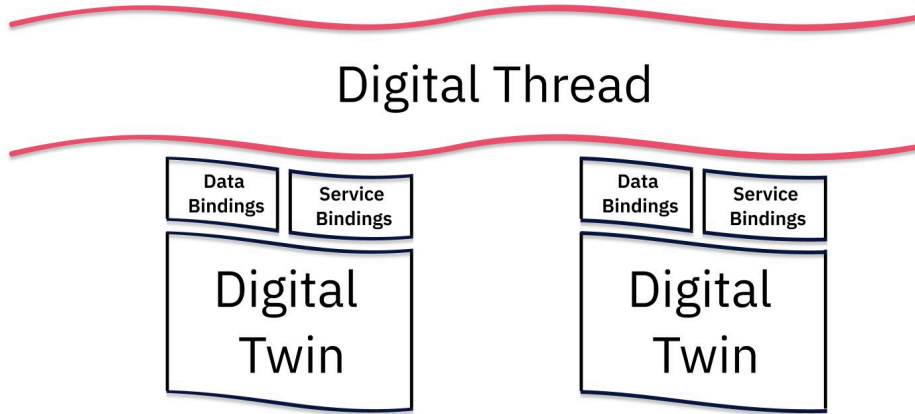
Subaru Telescope



EMIT Spectrometer Lead ISS Instrument



SysML v2 as Digital Thread Language



```
package SpecAndAnalysis {  
  private import ISQ::*;  
  private import SI::*;  
  
  part sc : Spacecraft;  
  
  requirement maxrequiredMaxFuelMass:MaxFuelMassRequirement {  
    doc /* the actual fuel mass shall be less than 0.25 kg*/  
    attribute :>> requiredMaxFuelMass:ISQ::MassValue = 0.25 [kg];  
  }  
  
  part analysisContext {  
    analysis fuelMassAnalysis : FuelMassAnalysis {  
      subject :>> sc :> sc;  
      in attribute startAltitude = 400 [km];  
      in attribute targetAltitude = 500 [km];  
      in requirement fuelMassRequirement = maxrequiredMaxFuelMass;  
    }  
  
    part mysc_analyzed :> sc {  
      attribute :>> fuelMass = fuelMassAnalysis.calculatedFuelMass;  
    }  
  
    satisfy maxrequiredMaxFuelMass by mysc_analyzed;  
  }  
}
```

Requirement

Analysis

Verification

Sysml is a lightweight language for constructing digital threads

From SysML v1 to SysML v2

AI-Assisted, Selective Model Migration

More teams can begin the SysML v2 journey now

Dassault Systèmes provides native SysML v1 → v2 migration in 2026x. Kotar + Kotari extend migration options to organizations continuing with Cameo 2022x or 2024x.

NATIVE PATH

**Cameo / CATIA Magic
2026x**



**Dassault native v1 → v2
migration**



SysML v2 models

EXTENDED PATH

**Existing Cameo
2022x / 2024x**



Kotar + Kotari migration assistant



SysML v2 models

Preserve engineering investments

Adopt v2 at your own pace

Built on OpenSysML technology

Three migration use cases

One toolchain supports wholesale conversion, interactive knowledge extraction, and deliberate restructuring—not just one-to-one translation.

01 MOVE THE MODEL

Full-model migration

Convert all supported SysML v1 content from Cameo 2022x / 2024x into a reviewable SysML v2 model.

Broad scope • **local conversion**

02 TAKE WHAT MATTERS

AI-assisted selective migration

Use Kotari to explore a legacy model, identify useful packages and relationships, then migrate selected content.

Natural language • **incremental**

03 TRANSFORM, THEN REFINES

Two-pass modernization

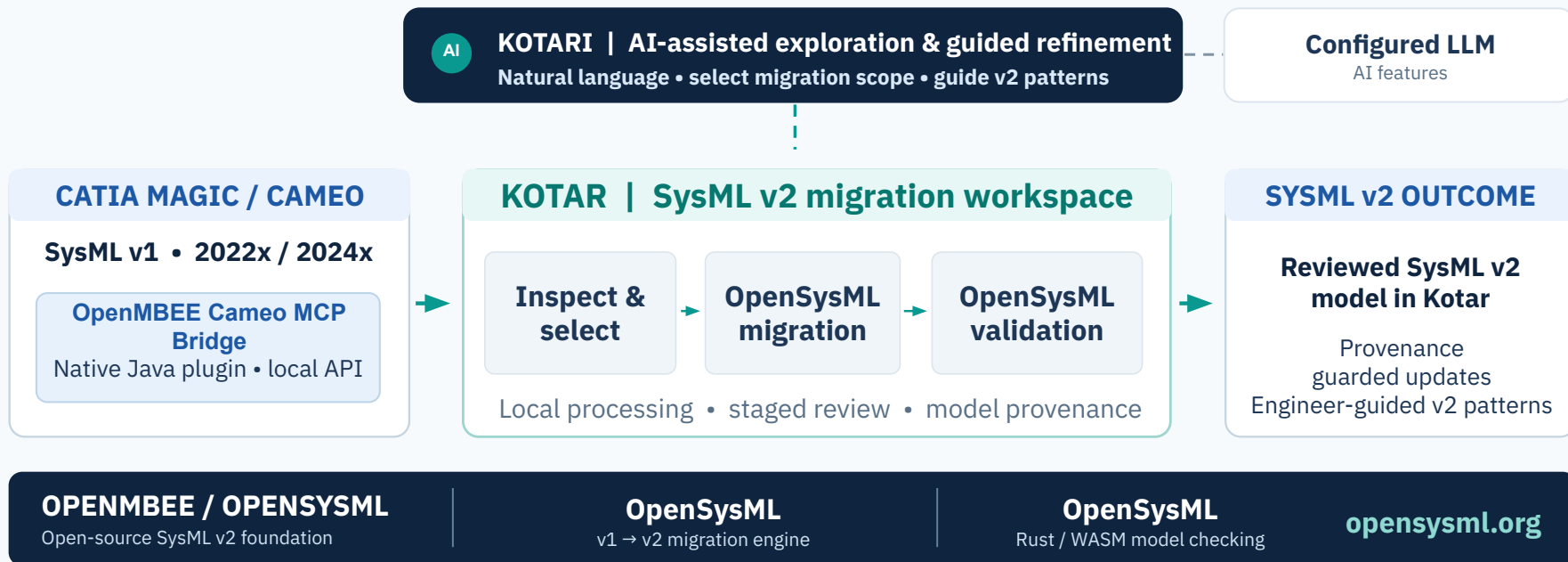
First migrate all or part of the model. Then restructure the raw v2 result using organization-specific SysML v2 patterns.

AI-guided refinement • **reviewed**

Across all paths: Semantic validation • Staged engineering review • Provenance • Guarded updates

Collaborative architecture: Cameo + OpenSysML + Kotar

Preserve existing Cameo investments while using open-source/free SysML v2 technology for AI-guided migration, validation and refinement.







Starforge Academy

- Training Curriculum
- User Community:
- OpenMBEE community
 - contribute and receive
- Paid Membership Community:
 - 1:1 time with Trainers
 - Guidance on Project Specific Questions

Trainer Bench



Tim Weilkens



Vince Molnar



Sandy
Friedenthal



Robert Karban



Hans Peter de
Koning



Ed Seidewitz

Training Courses:

- Introduction to **SysML v2**
- Advanced **SysML v2** Modeling
- Geometry Integration with **SysMLv2**
- **AI-Assisted MBSE** mit **SysMLv2**
- An Introduction to MBSE with **OOSEM v2**
- **OOSEM v2 Modernization Workshop**



Training Certifications!

© 2026 Planetary Utilities - All Rights Reserved

Trainer Bench



Tim Weilkens



Vince Molnar



Sandy
Friedenthal



Robert Karban



Hans Peter de
Koning



Ed Seidewitz

Object Oriented Systems Engineering Method

- OOSEM is an MBSE method that provides a top-down, scenario-driven process to analyze, specify, design, and verify a system
 - System model is a primary artifact
 - Controlled as part of the technical baseline
- Provides a way to deal with system design and development complexity using separation of concerns as a fundamental tenet
 - Black box, white box
 - Logical, physical
 - Distribution of functionality, data and control
 - Operational system, enabling systems
- Should be tailored to satisfy project objectives and constraints

Workshop Objectives

- Identify potential **improvements to the engineering/enterprise** workflows
 - Focus on workflows used to create products
 - Identify MBSE methods that can be leveraged to improve the workflows as appropriate

Workshop Approach

- This is a **working session to engage stakeholders** involved in the **engineering/enterprise modernization project** (engineers, project team, leadership, others)
- A systematic process based on OOSEM will be used to **facilitate the discussions and identify improvements**
 - As-is
 - To-be
 - Project and transition planning
- The work will be done in **group discussions and a series of breakout sessions**
- An unclassified example will be used as a representative example to enable open dialogue and information sharing, where applicable

Desired Workshop Outcomes

- **Shared understanding** of how engineers fit into the broader enterprise
- Common **cross-organization** engineering workflow **needs**
- **Opportunities** for improved engineering workflows, automation, and integration
- High-level improvement **roadmap**

