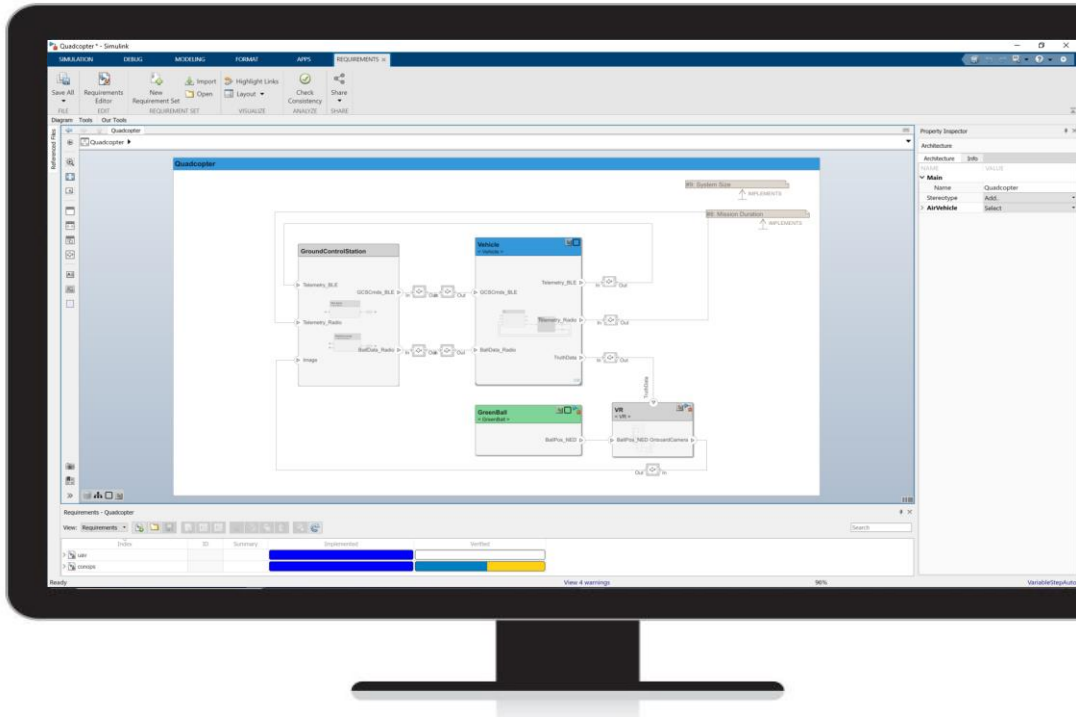
Design and Analyze a Quadcopter with Model-Based Systems Engineering (MBSE)

Cori Harber
Application Engineer
MathWorks – Natick, MA

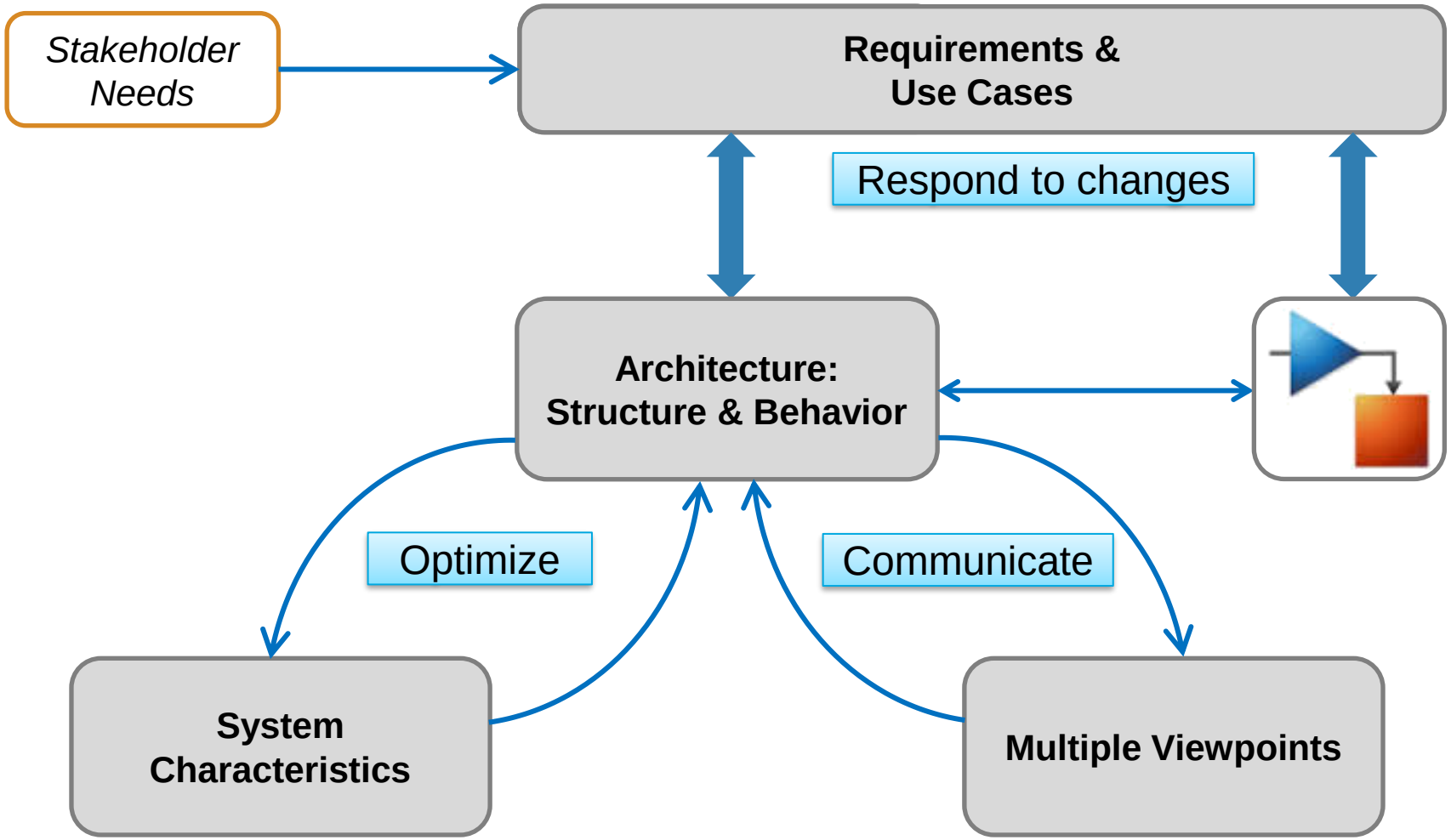


Agenda

- Model-Based Systems Engineering in the MathWorks Ecosystem
- Overview of System Composer and Getting Started
- Interoperability

Model-Based Systems Engineering in the MathWorks Ecosystem

Typical System Engineering Workflow



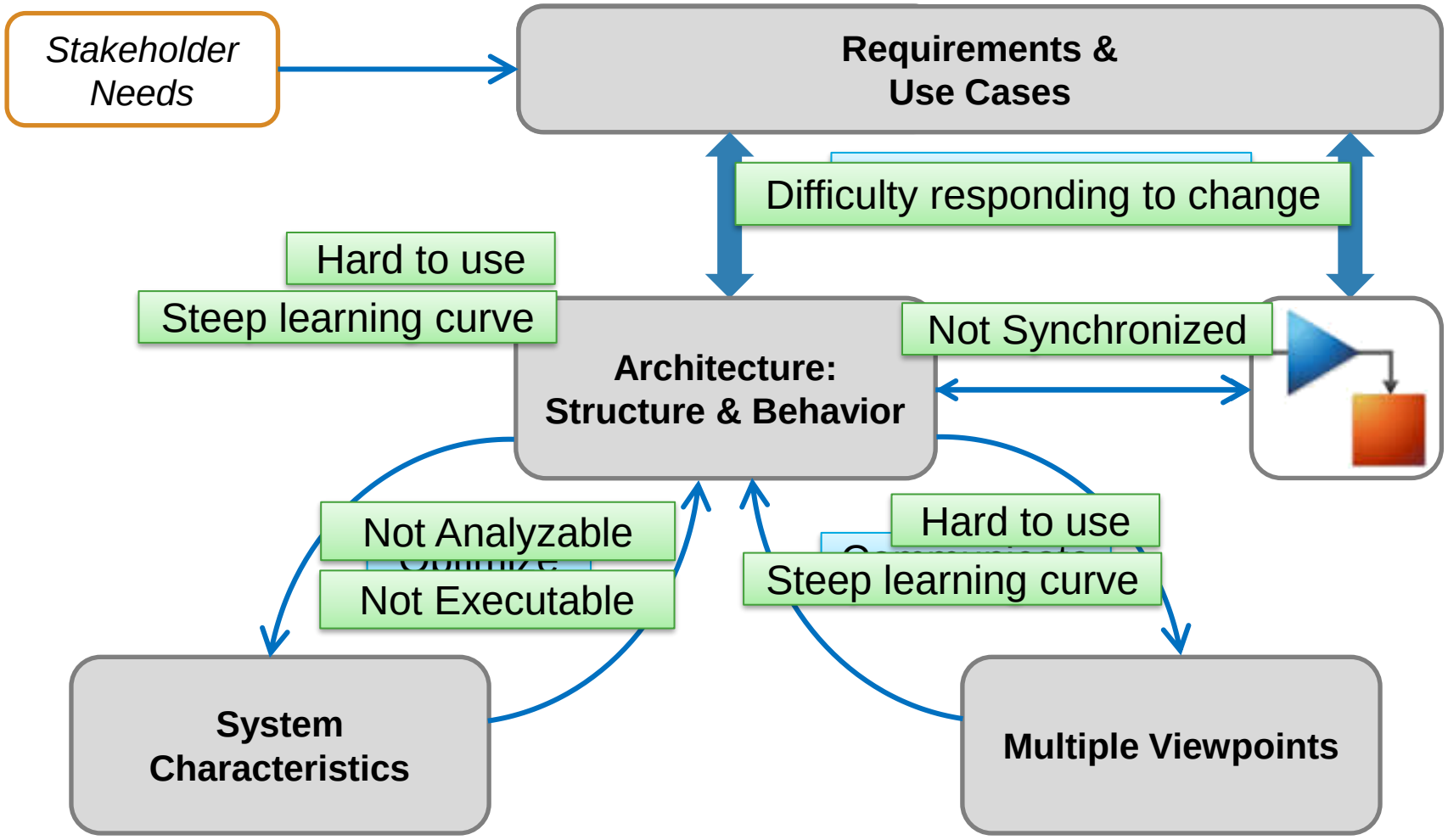
Highly Iterative

Highly Collaborative

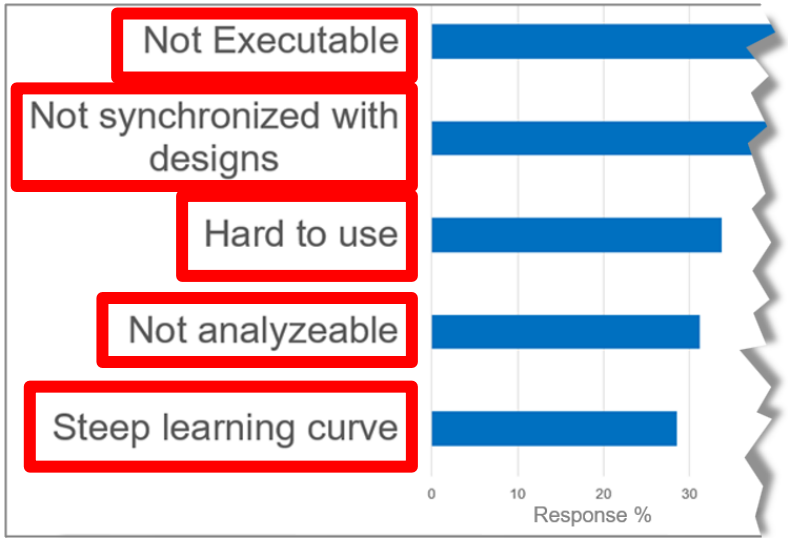
Deliverables:

- Specifications
- ICDs
- Reports
- Code
- More….

Challenges in a System Engineering Workflow



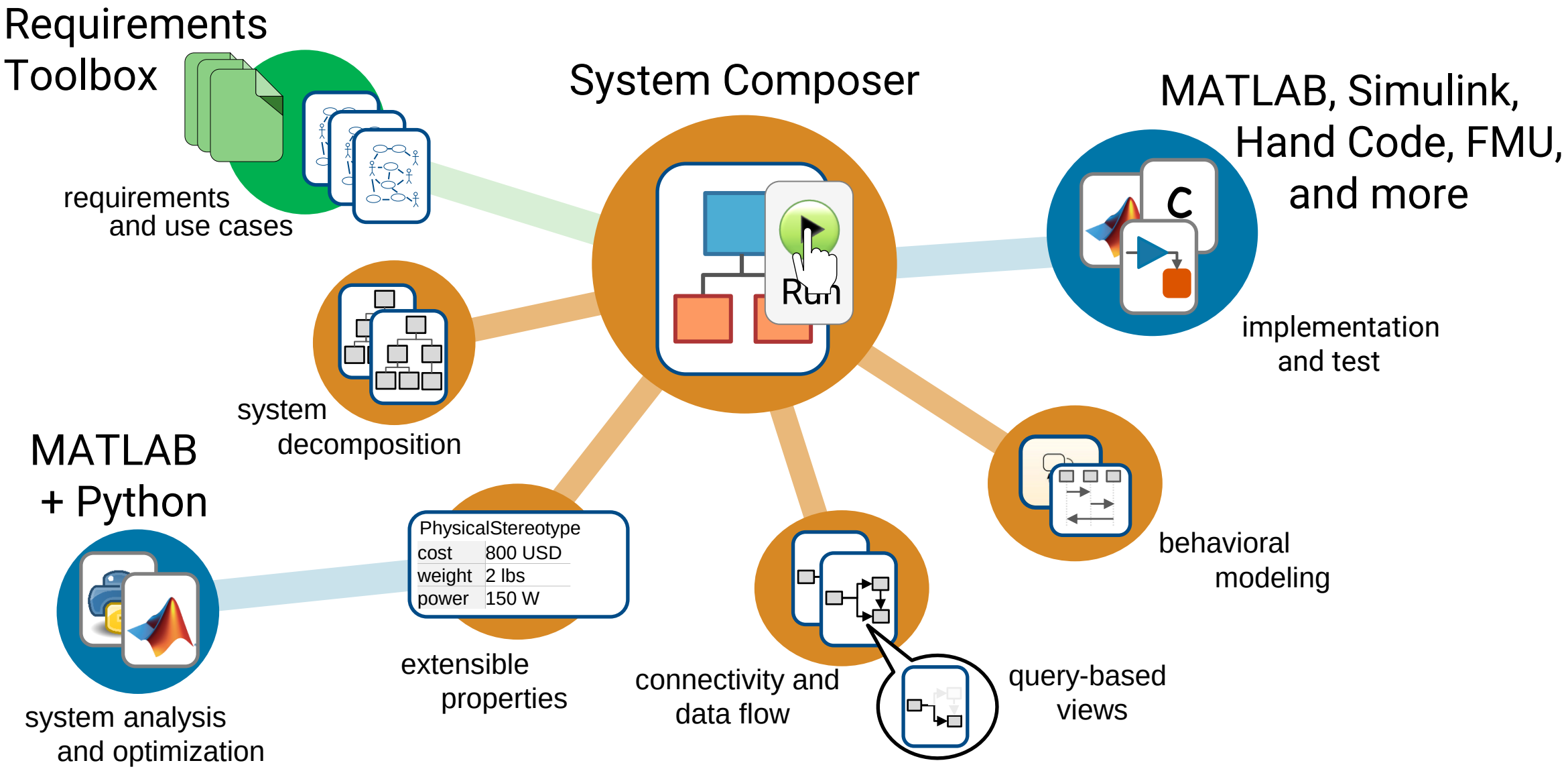
Common Pains



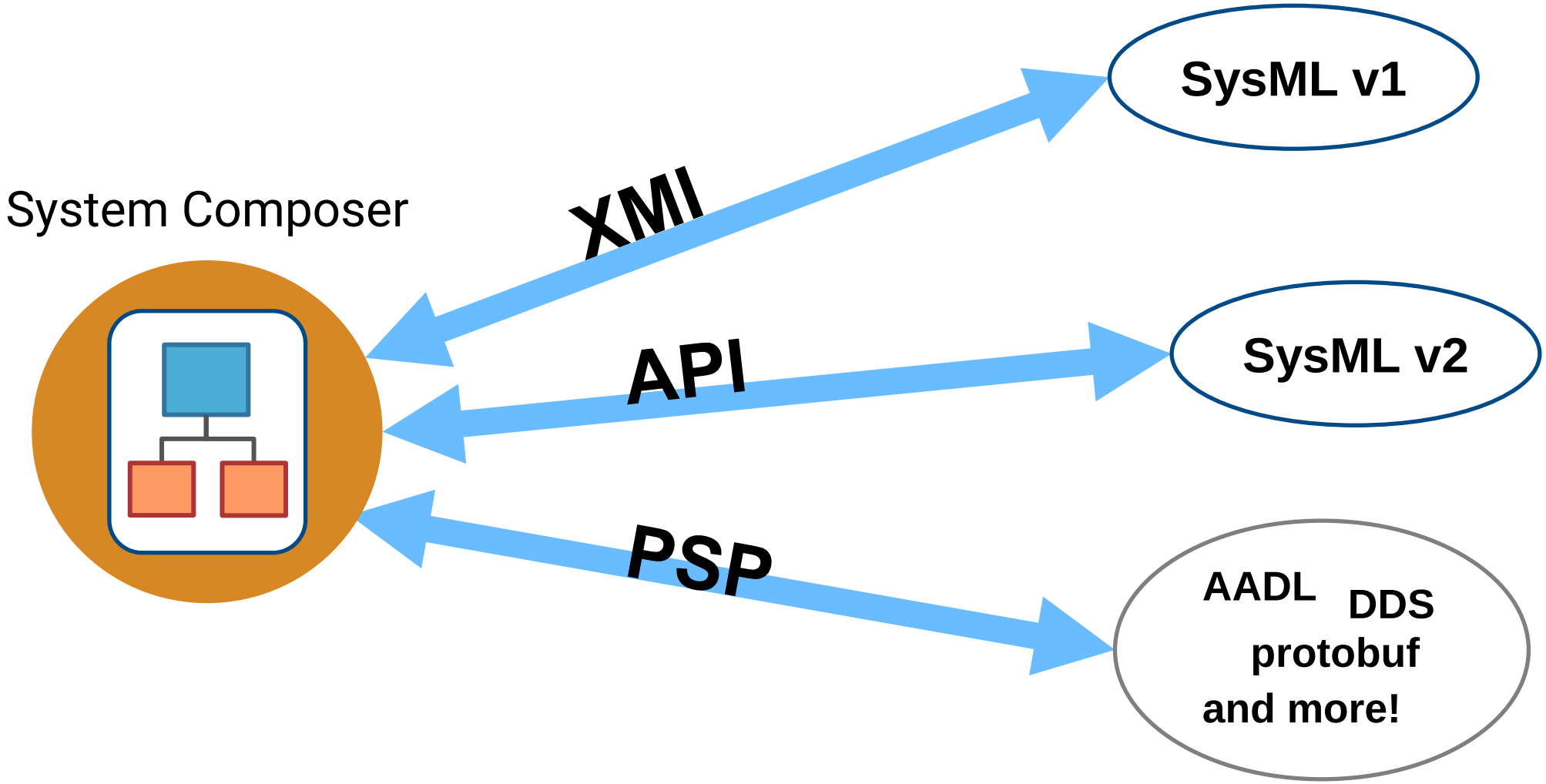
Deliverables:

- Specifications
- ICDs
- Reports
- Code
- More….

Model-Based Systems Engineering at MathWorks



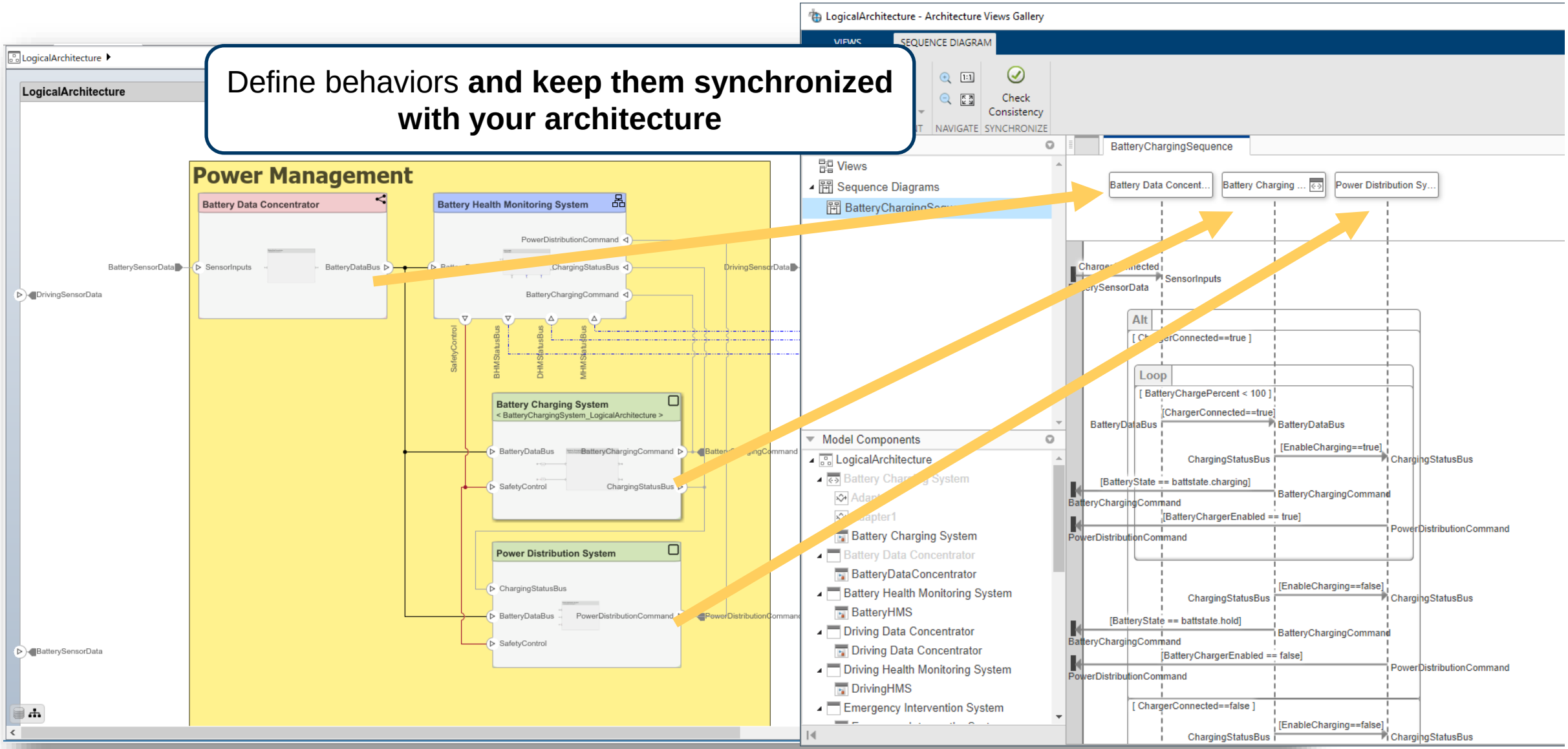
System Composer supports multiple languages



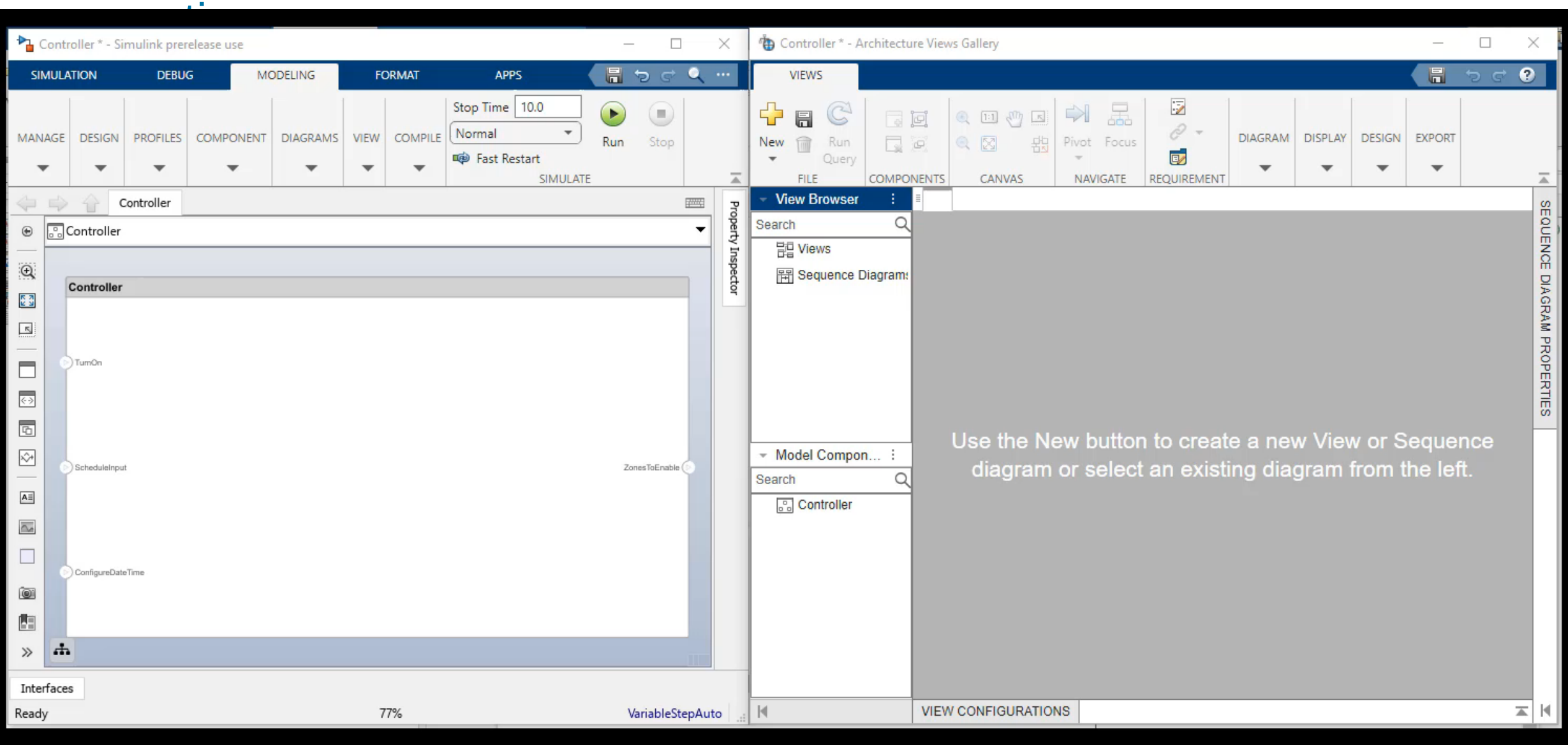
Overview of System Composer and Getting Started

Capture Intended Behavior with Sequence Diagrams

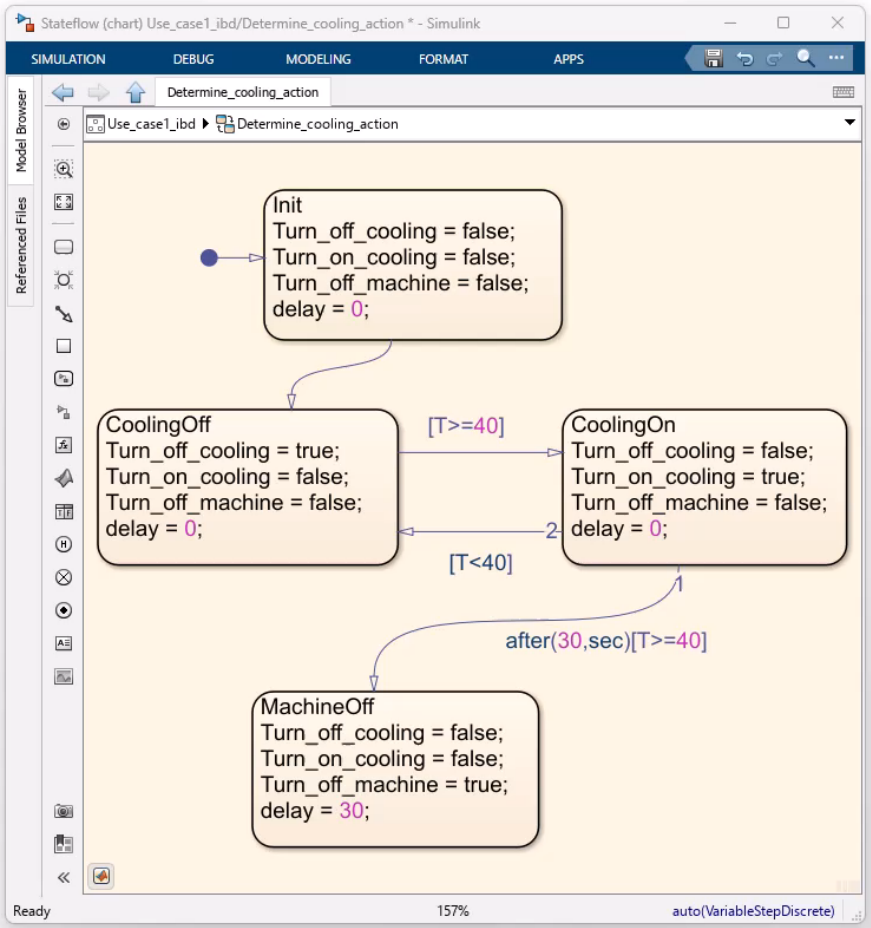
Define behaviors **and keep them synchronized with your architecture**

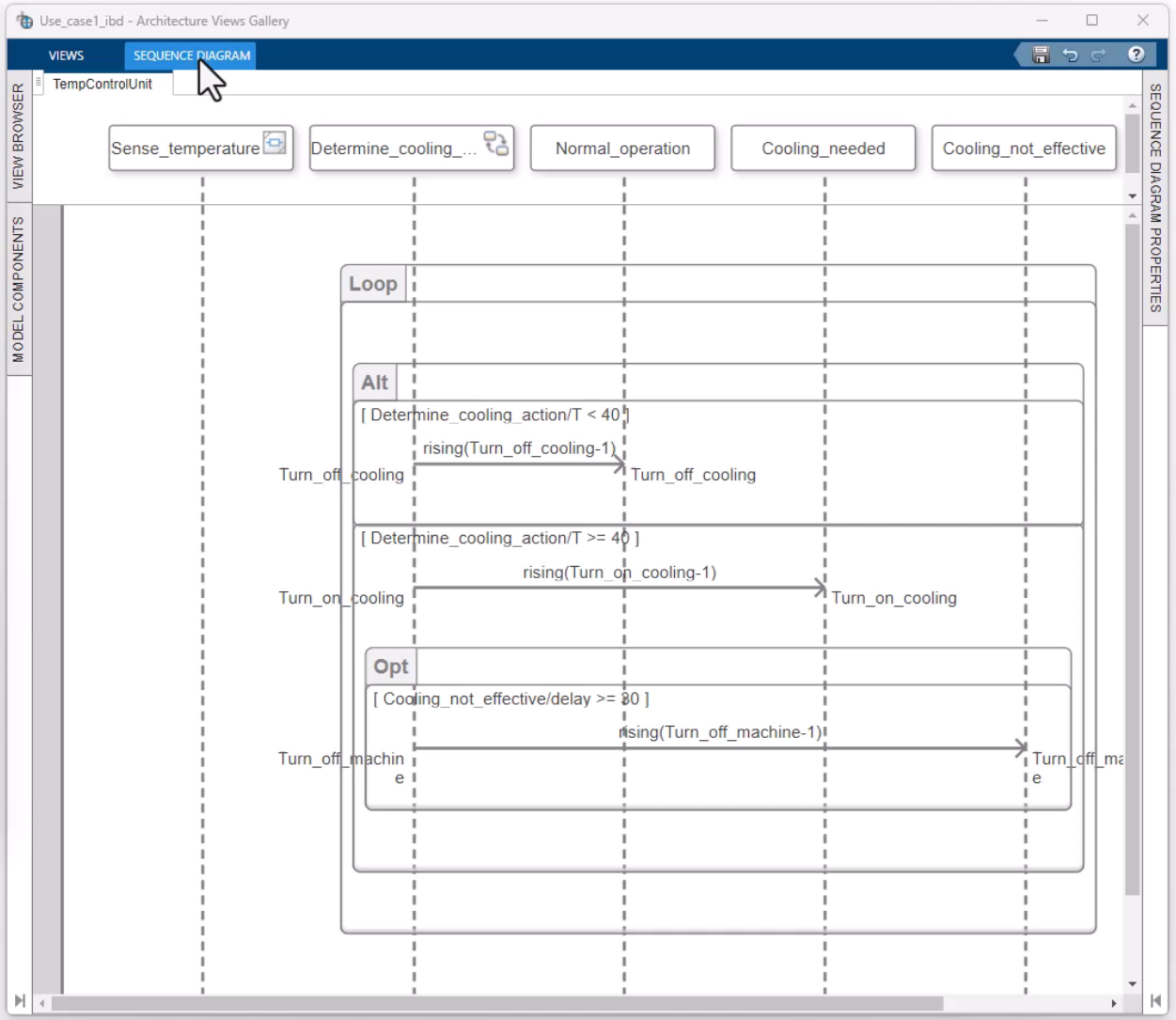


Describe your system components while specifying system

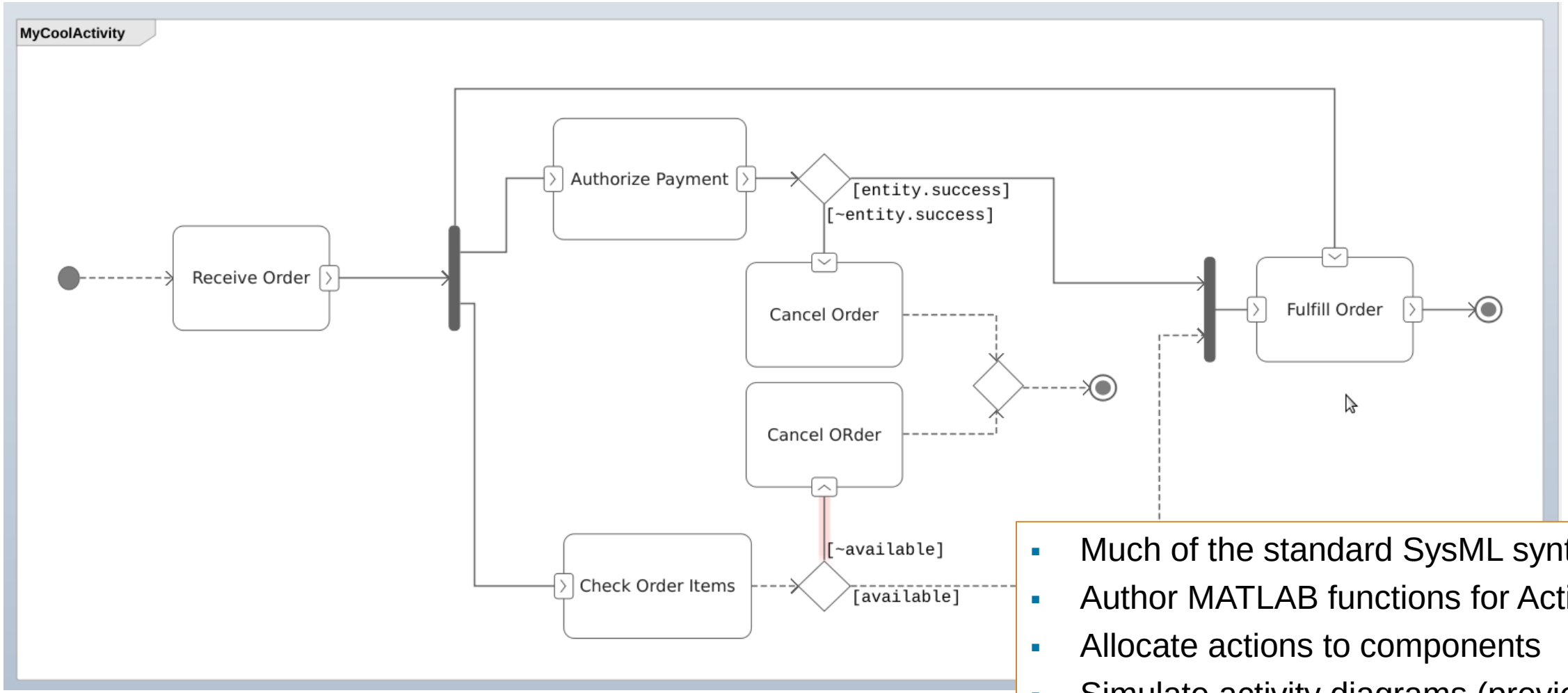


Verify Behavior of Detailed Design Models





Author Activity Diagrams to specify system behaviors, functional architectures or scenarios



- Much of the standard SysML syntax
- Author MATLAB functions for Actions
- Allocate actions to components
- Simulate activity diagrams (preview)

Interoperability

Interchange with SysML v1 models

SysML Connector

Import SysML models into System Composer

Request support package

Import your SysML v1 models into System Composer.

Export your System Composer models to SysML v1.

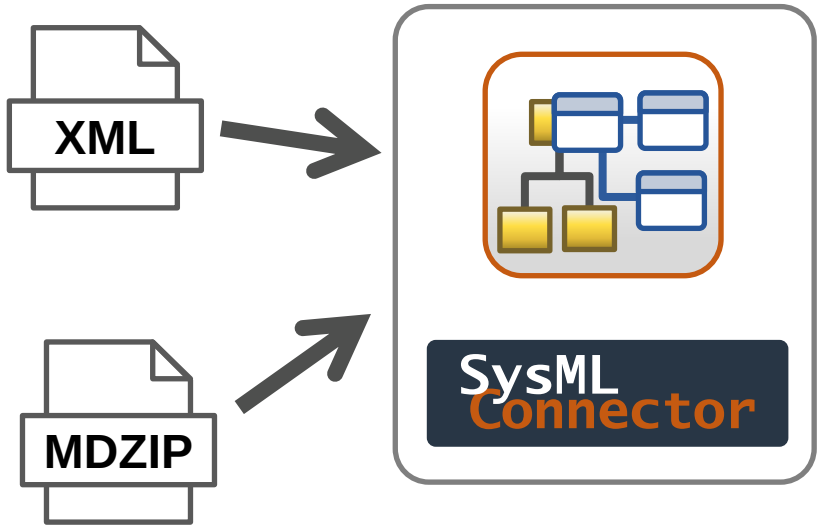
- Limited to architectures and interfaces
- We are investing in a SysML v2 solution to replace this

To request the SysML Connector:
https://www.mathworks.com/products/sysml.html

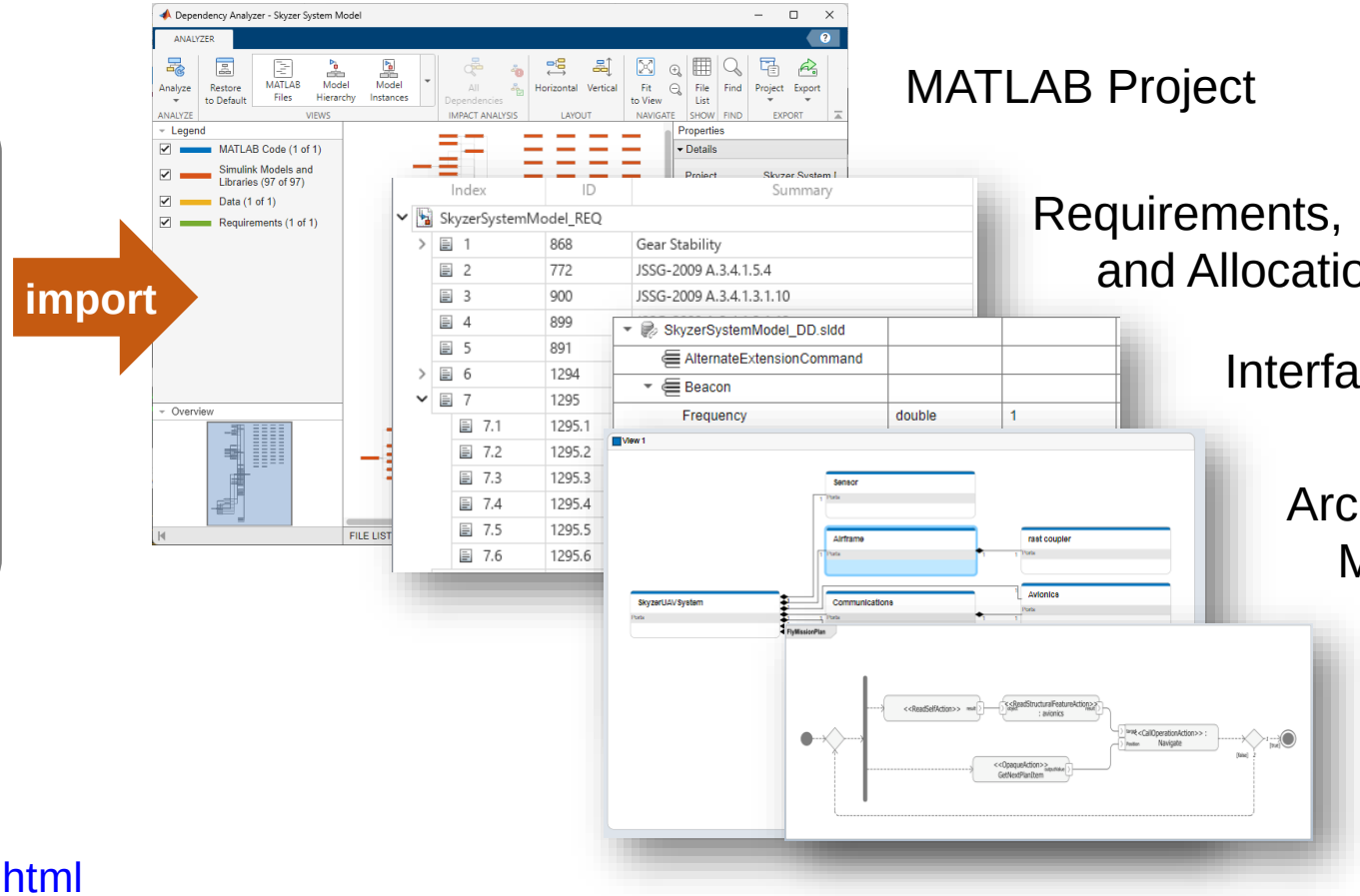
Where do I start?

- Use the SysML Connector to import your SysML v1 model into MATLAB

SysML v1 Artifacts



MathWorks Artifacts



MATLAB Project

Requirements, Links, and Allocations

Interfaces

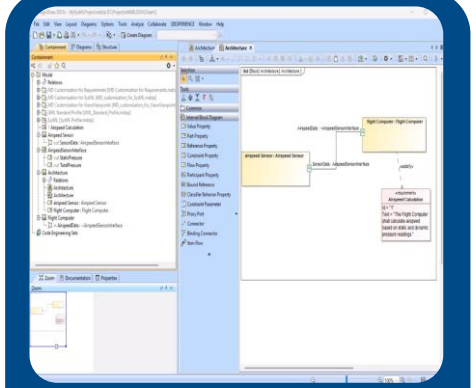
Architecture Models

Activity Diagrams

To request the SysML Connector:
https://www.mathworks.com/products/sysml.html

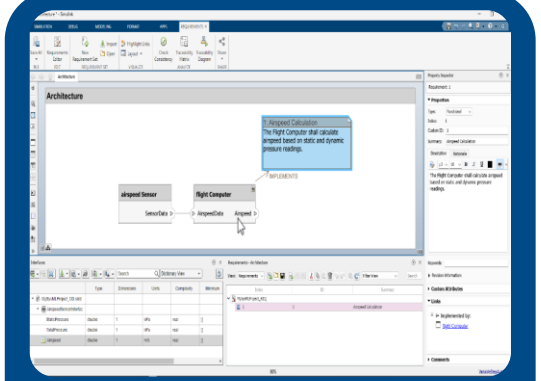
Example Use Case

import



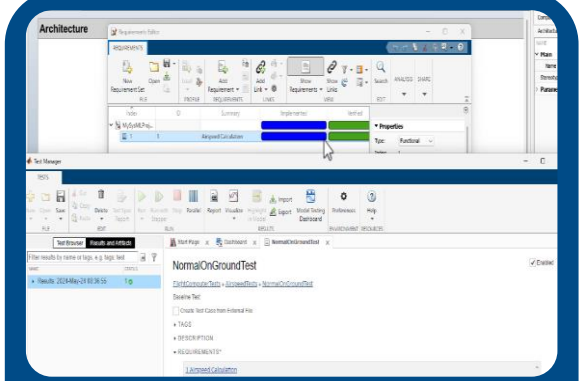
SysML v1 Model

refine



System Composer
and
Simulink

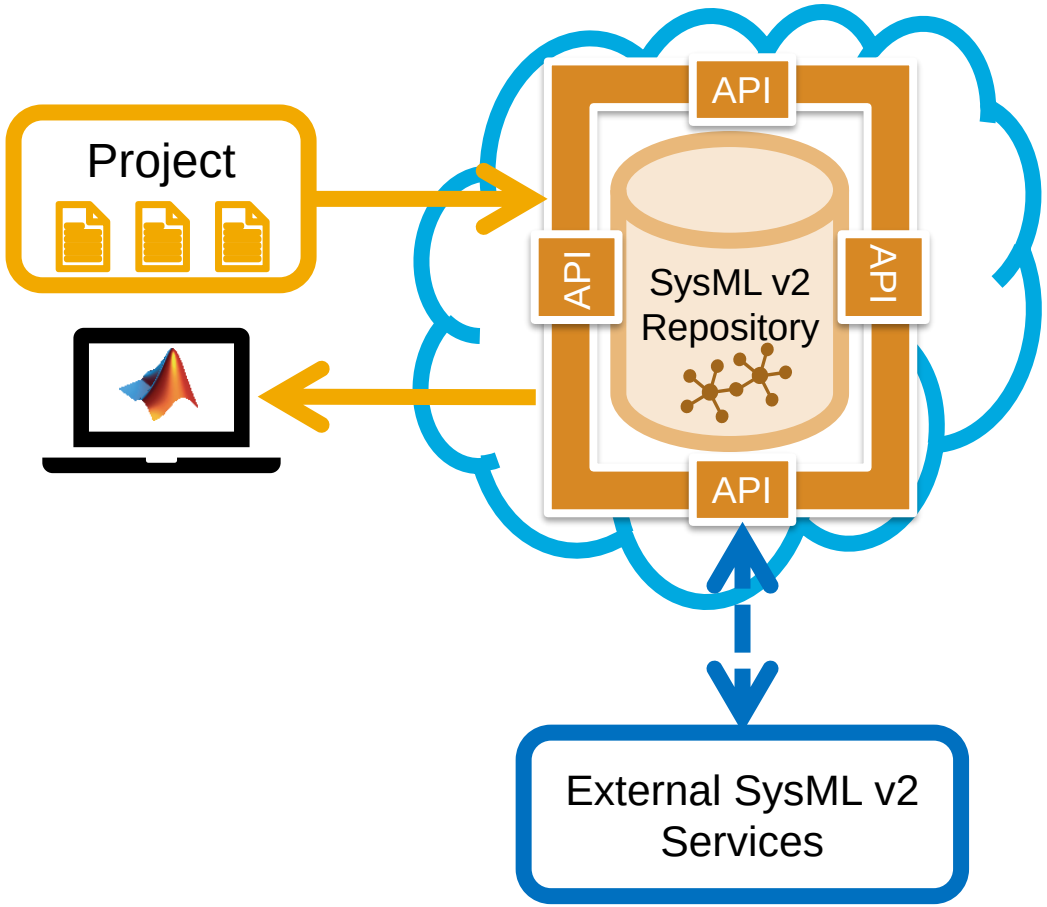
verify



Requirements Toolbox
and
Simulink Test

MathWorks is preparing to support the SysML v2 standard

- Work in System Composer today and be ready for SysML v2 tomorrow!
  - System Composer is well-aligned with the concepts of SysML v2
- Interoperability is our top priority.
  - We plan to provide access to System Composer model data through SysML v2 RESTful APIs
- Contact us to learn more about our beta later this year.



To request joining the beta program:
https://www.mathworks.com/products/sysml.html

SysML v1 Migration Service

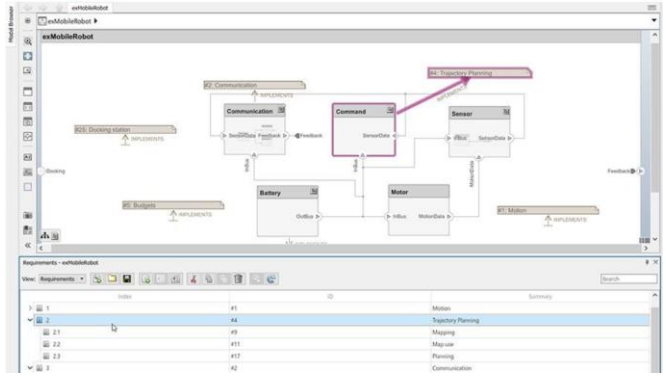
Challenge: Transition to System Composer

Thousands of models and design artifacts built in SysML v1 tools. The MBSE community is moving to v2, requiring migration to new tools or loss of years of engineering work.

Solution: MathWorks Consulting Services and System Composer

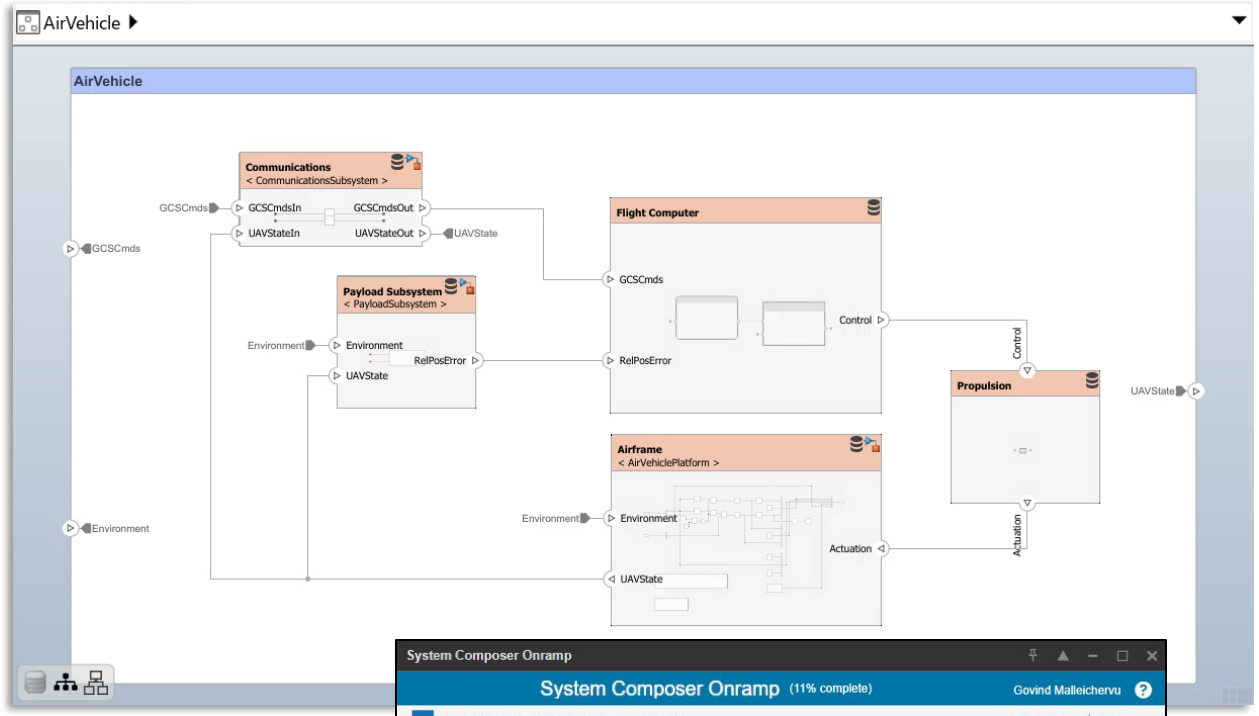
MathWorks Consulting will:

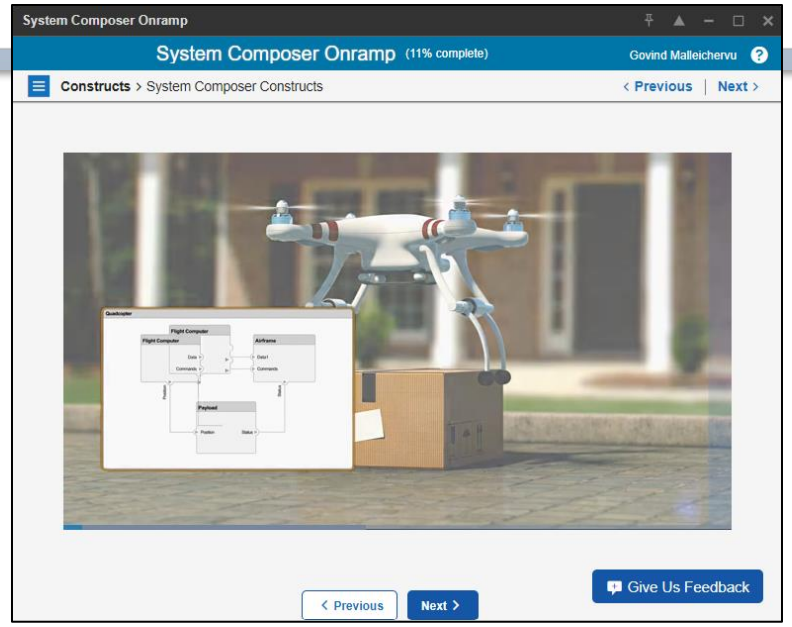
- Analyze your SysML v1 assets and your needs.
- Create and execute a custom plan to migrate models and data to a future-ready environment based on MATLAB and System Composer.



System Composer Onramp

- Learn how to perform **model-based systems engineering** by creating simulatable architecture models using System Composer in 2 hours
- **Free** to Everyone
  - Access online or from Simulink (starting with R2024a)
- Short video demonstrations and hands-on exercises with immediate feedback
- Learn by **Doing**:
  - Build a descriptive architecture
  - Generate views with filters
  - Link requirements to the architecture
  - Elaborate the architecture with behavioral models
  - Simulate and test the architecture





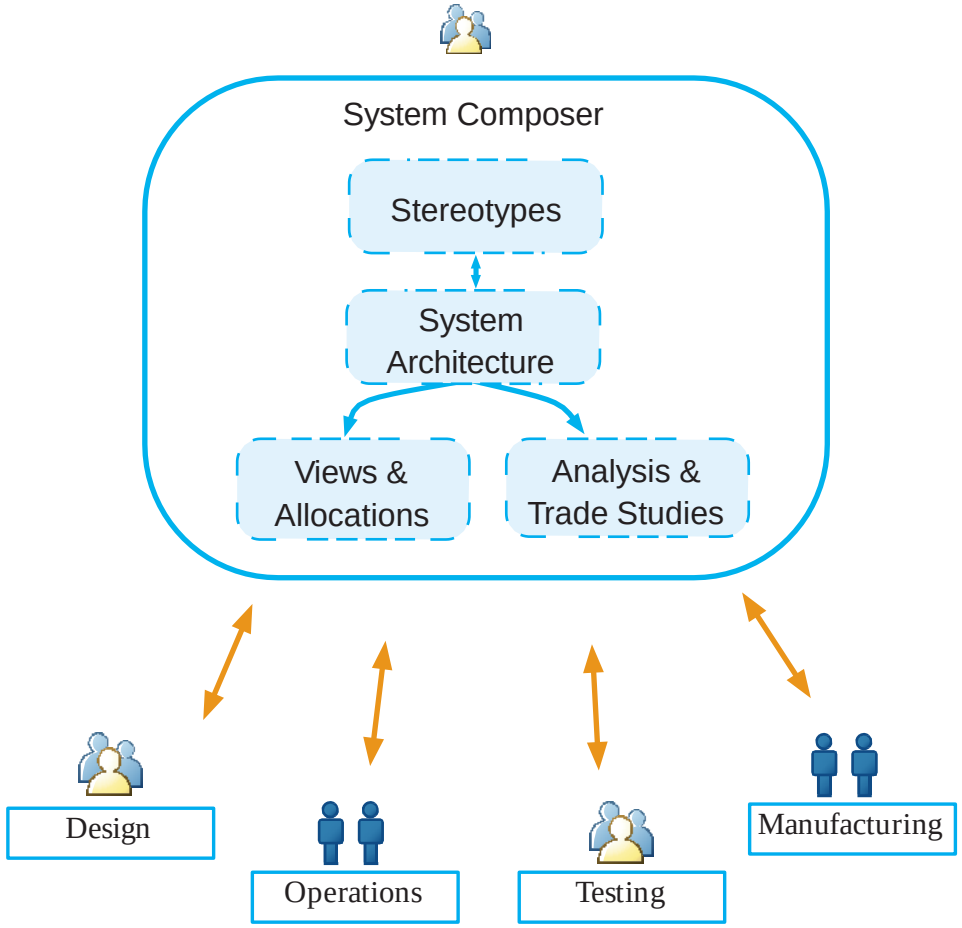
Self-Paced

System Composer for Architecture Modeling

After this 1-day training you will be able to:

- Identify the role of architectural modeling in Model-Based Design
- Create functional, logical, and physical architectures
- Customize existing components, ports and connections
- Analyze architecture models for early requirement validation
- Generate customized architecture visualizations

[See detailed course outline]



Learn More

- Solutions: Model-Based Systems Engineering
- Products:
  - System Composer
  - Requirements Toolbox
  - Simulink Test
  - Simulink