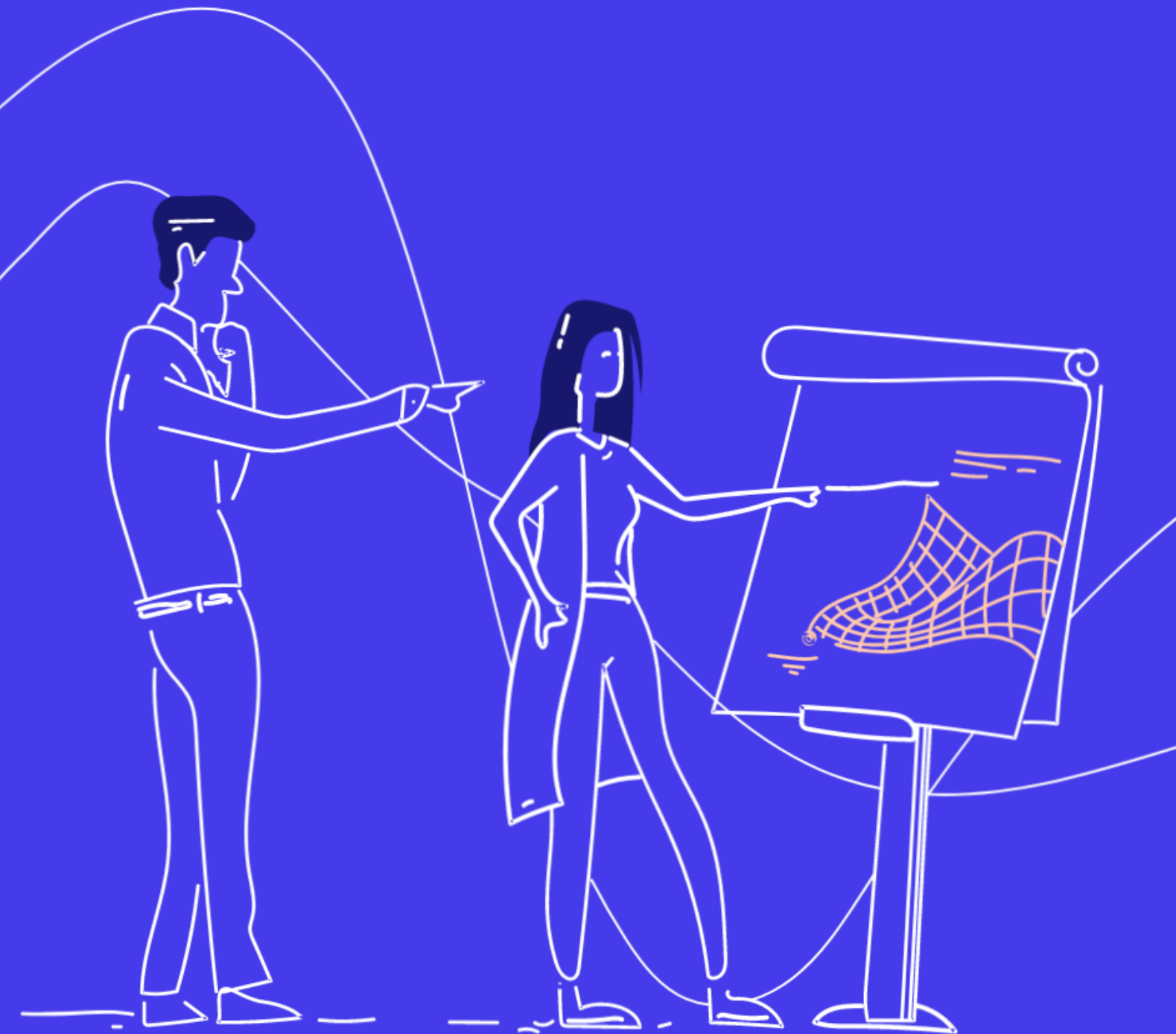




System Composer™ for Architecture Modeling



SciEngineer's training courses are designed to help organizations and individuals close skills gaps, keep up-to-date with the industry-accepted best practices and achieve the greatest value from MathWorks® and COMSOL® Products.

System Composer™ for Architecture Modeling

This one-day course focuses on developing and analyzing model-based architectures with System Composer™ and Requirements Toolbox™.

Prerequisites

Simulink Fundamentals or equivalent experience using Simulink is needed. Basic knowledge of Systems Engineering concepts is recommended.

DURATION	LEVEL
1 Day	Advanced
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>

TOPICS

Day 1

- Introduction to Model-Based Design
- Working with Requirements
- Logical System Architectures
- Physical System Architectures
- Capturing and Analyzing System Properties
- Enhancing Architecture Traceability

Introduction to Model-Based Design

OBJECTIVE: Introduce the concept of Model-Based Design and outline the advantages of System Composer.

- Advantages of System Composer
- Model-Based Design Process
- Subsystem Implementation
- Testing, Verification and Validation
- Course Example Walkthrough

Working with Requirements

OBJECTIVE: Articulate general attributes of requirements and capture requirements with Requirements Toolbox.

- Identifying and writing high-level requirements
- Writing requirements
- Creating requirement sets
- Importing requirements

Logical System Architectures

OBJECTIVE: Introduce the three main types of system architectures and build logical architectures.

- Distinguishing between functional, logical and physical architectures
- Creating logical architectures
- Creating interfaces in System Composer
- Linking Simulink models

Physical System Architectures

OBJECTIVE: Discuss how to create variant and reference components, and how to use physical ports.

- Creating physical ports in physical architectures
- Adding reference architectures
- Adding variant components

Capturing and Analyzing System Properties

OBJECTIVE: Capture system properties for individual elements with stereotypes and leverage MATLAB® to analyze and aggregate these properties.

- Introducing profiles and stereotypes
- Creating and customizing stereotypes for elements
- Creating stereotype properties
- Analyzing system properties with MATLAB

Enhancing Architecture Traceability

OBJECTIVE: Generate views, diagrams, allocations, and tracking requirements to enhance the digital thread of the project.

- Generating hierarchical diagrams of architecture models
- Highlighting specific paths, domains, and areas of responsibility in architecture models
- Using sequence diagrams
- Allocating between functional, logical, and physical architectures
- Tracking requirement traceability



**Expand your
knowledge**

