

Systems Engineering using SysML

Course category	SysML
Training area	Design Techniques
Course code	SE-401
Duration	4 days
Additional information	For price: Contact Feabhas (available for on-site delivery only)

In today's increasingly competitive marketplace, many organisations are turning to systems engineering practices to improve their responsiveness to the needs and expectations of their stakeholders.

The SysML was defined as a subset and extension of UML2.0 designed to meet the needs of Systems Engineers. The aim is to produce a unified language for describing systems.

This four day course introduces the fundamental practices of Systems Engineering, using the SysML as its notation.

It covers the techniques and methodologies of Systems Engineering in a real-time embedded environment using SysML 1.5. The course is a mixture of lectures and practical exercises.

Course objectives:

- To understand the importance of Systems Engineering to a project
- To present the core concepts of Systems Engineering
- To give an overview of the Systems Engineering process
- To introduce the SysML notation

Delegates will learn:

- Requirements analysis techniques
- How to define system architectures
- Modelling system dynamic behaviour
- Techniques for analysing system performance
- The SysML notation

Pre-requisites:

Working knowledge of UML is useful, but not essential.

Who should attend?

The course is designed for Systems Engineers who wish to use the SysML in defining their systems.

The course is also recommended for engineers and managers who are responsible for delivering systems with high software content.

It can also be beneficial for systems process engineers

Duration:

- Four days

Course materials:

- Delegate Handbook

Course workshop:

During the course, delegates will participate in individual and group exercises to illustrate and reinforce lecture material.



Introduction to Systems Engineering

- What is Systems Engineering?
- Concepts of SE
- 'Cradle to grave' engineering
- The Systems V-Model

Differences between UML2.0 and SysML

- SysML as a subset of UML2.0
- Extensions to UML2.0

Requirements Analysis

- 'Traditional' vs. Viewpoint-oriented requirements
- Requirements Diagram
 - Modelling 'traditional' requirements
- Use Case model
 - Use case levels
 - Goal-oriented use cases
 - Defining Use Case Descriptions
- System Context Model
- System Modes model

Defining System Structure

- Package Diagram
- Block Definition Diagrams
 - Defining system logical structure
- Internal Block Diagrams
 - Defining internal architecture
 - Comparison with UML2.0 Structure classes
- Ports and Flows
 - Standard Ports
 - Atomic Flow ports and Item flows
 - Non-Atomic Flow ports and Flow specifications
 - Interfaces

System Dynamics Modelling

- Sequence Diagrams
- Activity Diagrams
- State Diagrams

Analysing system performance

- Constraints modelling
 - Parametric models
 - Allocating mathematical models to system elements

Standards

- IEEE-1471-2000
- ARP-4754

The Pragma+ process for SE

- Putting it all together

Feabhas Ltd - UK Office 15-17 Lotmead Business Park, Wanborough, Swindon, SN4 0UY, UK ·

+44 (0) 1793 792909 · info@feabhas.com · www.feabhas.com