

The Unified Modeling Language (UML®) Using Enterprise Architect

Three Day Course Syllabus

DAY 1

Introduction to Enterprise Architect

User Interface

- Desktop layout
- Portals window
- Visual styles
- Commonly used Views
- Workspace Layouts, Menu Sets, and Perspectives

Managing Projects

- Creating and opening Projects
- Creating Root Nodes, Views and Packages
- Adding Models using the Wizard
- Organizing Models with Views and Packages

Managing Diagrams

- Toolbox
- Creation techniques
- Menu options, properties and toolbar
- Navigation between diagrams
- Floating diagrams
- Copy/paste across diagrams

Creating Diagram Elements

- Creation techniques
- Properties
- Drag and drop

Managing Diagram Elements

- Appearance, layering, and feature visibility
- Alignment, resizing, automatic layout
- Info view

Deleting Model Elements

- Diagram level deletions
- Repository level deletions

Managing Connectors

- Creation (toolbox and Quick Linker)
- Redirection
- Advanced options
- Line bends and styles
- Virtualizing Connector ends

Managing Package Content

- Package Navigator
- Package Browser/List View

Tool Configuration

- Defining People
- Defining general types
- Personal configuration options

COURSE LEADER

Frank Truyen

is a principal consultant and trainer, with 20+ years of experience in the IT industry as a developer, architect, consultant and manager.

Strong expertise in different modeling notations such as UML®, SoaML™, DDS™, UPDM™, ArchiMate® and BPMN™, allied with his extensive experience in using the Enterprise Architect modeling tool, allows Frank to successfully provide training and consulting services to a broad variety of customers across many industries.

Modeling Business Architecture

Managing Requirements

- Creating Requirements via a diagram
- Creating Requirements using the Specification Manager
- Other features
- Importing & exporting Requirements
- Organizing Requirements

Structural models

- UML (Conceptual) Class diagram

Behavioral models

- UML Use Case diagram
 - Definition
 - Best practices
 - Creating Use Case diagrams
 - Discovering Actors
 - Guidelines for creating Use Cases
 - Modeling Use Case Scenarios
 - Defining constraints
 - Using behavioral diagrams to illustrate Scenarios
 - Tracing Requirements to Use Cases
- UML Activity diagram

DAY 2

Modeling Application Architecture

Structural models

- UML Class diagram (Attributes)
 - Object-Oriented principles
 - Classes and Objects
 - Element visibility
 - Relationships
 - Defining Attributes
 - Adding rules and constraints
 - Stereotypes and Tagged Values
 - Grouping elements into Packages
- UML Composite Structure diagram
- UML Component diagram
- UML Deployment diagram

Behavioral models

- UML Class diagram (Operations)
 - Discovering and assigning responsibilities
 - Patterns for assigning responsibilities
 - Defining Operations
 - Defining Interfaces
- UML Sequence diagram
 - Combined Fragments
- UML State Machine diagram
- UML Interaction Overview diagram

DAY 3

Traceability and Navigation

Overview

The Relationships window

The Relationship Matrix

The Gap Analysis Matrix

The Traceability window

Other traceability features

Core Tool Features

Documentation Generation

Template driven RTF generator

Diagram and Package options

Linking into Microsoft Word

Model documents

Master documents

Generating to an Artifact

HTML generator

Document Artifacts

Creating

Linked Documents

External document links

Model Searches

Initiating

Configuring

Viewing results

Custom searches

Other features

Managing Baselines and comparing models

Creating baselines

Administering baselines

Package comparisons

Diagram comparisons

Extra Tool Features

Diagram features

- Boundaries, Swimlanes, Matrix, and Kanban
- Content filters
- Notes
- Active legends

Element features

- Browser
- Compartments
- Replication

Tool features

- Working Sets
- Charts, Dashboards, and Heatmaps
- Managing hyperlinks
- Keyboard shortcuts
- Getting help

End-to-End Modeling Exercise (if time permits)

- The Online Bookstore model, or
- A User defined modeling exercise