

UML in Context

The Unified Modelling Language™ (UML) is the Object Management Group's (OMG) most-used specification, and is used widely to model not only application structure, behaviour, and architecture, but also business processes and data structure. The OMG has been an international, open membership, not-for-profit computer standards consortium since 1989 and whose board of directors includes representation from almost all organizations that shape enterprise and internet computing today.

UML stands for Unified Modelling Language and it provides a notation that is readily understandable by all stakeholders of the business, including the business analysts who define the system requirements, the developers who implement the software system and even the client, with a little support, who can readily understand much of the UML model of their proposed system.

UML began life in the mid 1990's as a language designed to support object orientated analysis and design, but has since expanded both in notation and use, to become the de facto language of choice for modelling every stage of the development process.

Stereotyping is a technique by which specific parts of the language can be extended to support particular disciplines. Eriksson and Penker's business analysis extensions will be of particular interest to the business analyst.

Summary

Duration: 3 days

This course will focus on the capture of business requirements to produce a business model. In a typical project process, this business model would then be handed over to the software designers to act as the foundation for their object orientated software design.

The course will feature Business Process Modelling, Use Cases, and Domain Modelling Workshops to establish the requirements of a complex business.

Objectives

By the end of the course attendees will:

- Be able to use the techniques supported by UML to produce a business requirements model
- Be able to construct Business Process Models, Activity Diagrams, Use Case Models, Domain (Class) Models and State Machines
- Have gained an understanding of the techniques used in Object Orientated Design using the UML Interaction Diagrams
- Be able to adapt the UML Interaction diagrams to general business analysis uses
- Have an appreciation of the function of all UML 2.0 diagrams

Prerequisites

Although the course can be readily adapted for delegates with little or no business analysis experience, the aim of the course is to effectively and efficiently equip business analysts, without UML experience, to be fully proficient in the use of the UML in their analysis work.

Those who have no business experience may prefer our Introduction to Business Analysis Using UML course, which spends more time focusing on Facilitated Workgroups, Interviewing Skills and other general business analysis techniques.



UML
Designed and Managed by
the Object Management
Group

Detailed Course Description

Introducing UML

- Models v Diagrams
- What UML is and what it is not
- Modelling Notations
- The Waterfall Lifecycle
- The Unified Process
- The Iterative Lifecycle

Object Orientation Basics

- Pros and Cons of Procedural Programming
- Pros and Cons of Object Orientation
- Classes
- Encapsulation
- Collaboration
- Object to Relational Mapping
- Persistence

Business Modelling

- Beginning the Analysis
- Business Model
- Business Processes
- UML Stereotypes
- Eriksson-Penker extensions
- Business Process Overview Diagram
- Business Actors
- Business Workers
- Case Workers

The Activity Diagram

- Activities, Sub-activities and Actions
- Linking the Activities
- Control Flow
- Object Flow
- Alternative Object Representation
- Initial, Final and Flow Final
- Forks and Joins
- Forks, Joins and Control Flow
- Decision Points and Merges
- Signals
- Objects and Signals
- Swimlanes
- Activity Partitions
- Interruptible Activity Region
- Pins
- Expansion Region
- Parameter Set

Use Cases

- Use Cases as structured requirements
- Granularity of Use Cases
- Uncovering Use Cases from Business Processes
- Primary and Secondary Actors
- Ranking Use Cases
- Specifying Use Cases
- Use Case Descriptions
- Non functional requirements
- Style Guidelines

Detailed Course Description Continued

The Domain Model

- Domain Modelling
- Finding Classes
- Notation in the UML
- Attributes
- Associations
- Multiplicities
- CRUD Matrices

The State Machine

- Capturing Business Rules
- Events and States
- Basic Notation
- Superstates and Substates
- Conditional Transitions
- Actions
- Finding Use Cases from the State Model

Detailed Use Case Descriptions

- Use Case Storyboards
- Preconditions
- Postconditions
- Main Flow
- Extension Flow
- Graphical Form

Interaction Modelling - The Communication Diagram

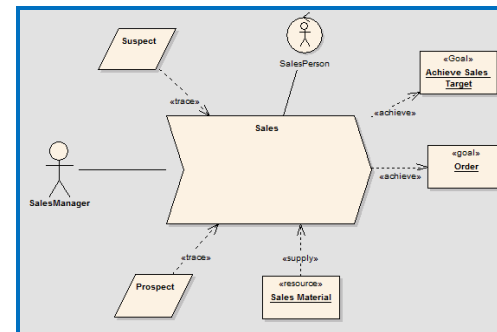
- Objects and Associations
- Method Calls
- Parameters
- Return Values
- Looping
- Creating Objects
- Mapping to the Design Class Diagram

Interaction Modelling - The Sequence Diagram



Use of all UML 2.0 Diagrams

- Behavioural Diagrams
- Structural Diagrams
- Interaction Overview Diagram
- Timing Diagram
- Component Diagram
- Composite Structure Diagram
- Deployment Diagram
- Object Diagram
- Model Driven Architecture (MDA)



Business Process Modelling

The UML has a huge vocabulary to provide great flexibility as to how we model process