Course description:
Modern embedded software design is constrained by an ever-growing set of constraints:
- Increasing software complexity.
- Decreasing time-to-market.
- The need to produce flexible, maintainable systems.
- The need to have rigorous engineering in critical systems.
These factors are driving the need for cohesive design methodologies and the use of modeling.

This is a detailed software design course which focuses on designing Real-Time Embedded Systems, using UML 2 notation to document the proposed design. The focus on design principles and methodologies make this course significantly different to most UML courses, which focus on notation.

Course objectives:
- To provide an understanding of Object Oriented design principles.
- To show how to develop real-time software in a rigorous and systematic manner.
- To enable attendees to develop their own practical design skills.
- To teach effective application of UML notation.

Delegates will learn:
- The fundamental concepts and terminology of real-time software.
- The diagrammatic and modelling underpinnings provided by UML for Object Oriented development.
- How to apply the design principles in real-time applications.
- The basics of an integrated, traceable and consistent approach in the development of software for real-time systems.

Pre-requisites:
- Some understanding of technical software development methods.
- Knowledge of typical embedded programming languages (like C) is useful.

Who Should Attend:
- Designers new to the area of real-time software design.
- Developers with some non-embedded UML experience.
- Designers embarking on projects using UML-based techniques for the first time.

Duration:
Five days

Course Materials:
- Delegate handbook
- All worked examples and solutions

Related courses:
- OO-101 An Overview of UML for Real-Time Embedded Systems
- OO-302 Software Modelling with UML
- SE-501 Real-Time Software Engineering
- SE-401 Systems Engineering using SysML
- DP-402 Design Patterns in C++ for Embedded Systems

Course Workshops:
Approximately 50% of the course involves practical application of the techniques discussed. Delegates work in small groups dealing with problems based on real-world systems.

The course specifically does not make use of a CASE tool. From our experience a CASE tool distracts delegates from learning design issues and UML. However, the workshops clearly demonstrate the benefits and disadvantages of CASE tools, thus aiding CASE tool selection.