

## Foreword

The notion of supporting software development using artificial intelligence techniques has been around for more than 20 years. In the 1980s, work on the Programmer's Apprentice project led to an 'intelligent' program editor and this led to a number of other projects that aimed to build an intelligent programming environment.

Although some of the work in these projects was exploited under other headings, the notion of an intelligent environment was never realised in practice. Sadly, the whole area of AI and software engineering was oversold and this set back the whole area for many years.

One reason for this was that relevant AI techniques are often computationally expensive and scaling these to apply to large software systems was simply impractical. Now, the limit on computation is disappearing and new hardware and software architectures mean that there is now a real possibility of making AI work effectively in the development of large and complex software systems.

This is an important book because it reflects the re-emergence of AI and Software Engineering as a valid and potentially very valuable research field. The key areas addressed are project management, requirements engineering, software design and software testing. These are all areas where our conventional approaches to software engineering are running out of steam as system sizes increase and industry demands higher quality, shorter times to deployment and lower costs. We see themes emerging that cross-cut these different areas such as the use of Bayesian networks and the encoding and utilisation of domain knowledge in tools to support software engineering activities.

The papers in this book represent an excellent summary of the state of the art in AI and Software Engineering. Most of the work described here is still at an early stage of development and there are surely many engineering challenges to be overcome before these techniques can be routinely used to support large system construction. The challenges to be addressed are to scale the techniques to larger systems and to demonstrate that these are cost effective in reducing costs and schedules and improving the quality of real systems. I believe that the work described here is a sound foundation for this engineering work and that the book should be read by all researchers interested in AI and Software Engineering and by far-sighted practitioners who are interested in the future of the discipline.

*Ian Sommerville  
St Andrews, Scotland  
March 2009*

***Ian Sommerville** is Professor of Software Engineering at St Andrews University, Scotland and was previously Professor of Computer Science at Lancaster University. He has been involved in software engineering research since the 1980s and his current research interests are in socio-technical systems and the dependability of large-scale enterprise systems. His textbook on software engineering was first published in 1982 and the 8th edition of this book was published in 2006. The text is used all over the world as a student text and has been translated into at least 10 different languages.*