Foreword

American Association for the Advancement of Science initiated the Project 2061 to systematically identify science, mathematics, and technology themes and provide education recommendations. The processes, observations, and recommendations are documented in “Science for all Americans,” published by Oxford University Press (ISBN 0-19-506771-1). In these wonderful guidelines, the “common themes” chapter starts like “some important themes pervade science, mathematics, and technology and appear over and over again, whether we are looking at an ancient civilization, the human body, or a comet. They are ideas that transcend disciplinary boundaries and prove fruitful in explanation, in theory, in observation, and in design.” Furthermore, the committee stated that these thematic ideas are SYSTEMS, MODELS, CONSTANCY AND CHANGE, and SCALE.

While we were developing a modern and unifying approach to advanced software engineering education at the University of Texas, in 1995 in Austin, we had also identified four key thematic ideas underlying the foundations of modern software engineering—the engineering and technology of 21st century. We also claimed that these topics should be covered as core elements of software engineering and moved to develop these contents into four courses. One of the editors of Modern Software Engineering Concepts and Practices: Advanced Approaches, Dr. Ali Dogru had been instrumental in the development of these four thematic areas. These areas were SYSTEMS (Systems in Project 2061), DESIGN (Models in Project 2061), PROCESS (Constancy and Change in Project 2061), and METRICS/MEASUREMENTS (Scale in Project 2061). As can be readily observed, these four core themes of software engineering also correspond to the themes identified by Project 2061 as “transcending disciplinary boundaries,” which is a testimony to the overarching comprehensiveness of modern software engineering beyond the common topics covered in popular software engineering books known as the software engineering body of Knowledge (SWEBOK). Therefore, I can safely claim that the book in your hand developed by two extraordinary software engineers and software engineering educators is not just another software engineering book regurgitating the material from SWEBOK and other related sources. This is a unique book looking into the complicated topic of software engineering with a fresh viewpoint.

If I may give a brief history of this development, I should start with our mentors Prof. Raymond T. Yeh and Prof. C. V. Ramamoorthy. Since the time of the establishment of IEEE Transactions of Software Engineering by Prof. Raymond T. Yeh in 1969 as the first software engineering journal and the subsequent expansion of it under Prof. C. V. Ramamoorthy’s chief editorship, software engineering went through several major evolutionary transformations. During the 1980s, due to rapid advances in the applications of software to many disciplines, the systems view started taking root. At the same time, basic software engineering research started moving from universities to corporations, which had been the fate for hardware research during the 1970s. We witness during the 1990s the explosion of networking capabilities
including Internet and World Wide Web, leading to the development of associated technologies related to systems integration. The first journal addressing to the issues of Systems Integration was initiated by our mentors Drs. Yeh and Ramamoorty again in collaboration with Dr. Peter Ng during early part of the 1990s. However, it was too early to introduce an applied systems integration journal since there have not been very urgent systems integration research problems yet. During 1995, Drs. Yeh and Ramamoorty, in collaboration with Dr. Herbert Simon and Dr. George Kozmetsky, spearheaded yet another development, the Society for Design and Process Science (www.sdpsnet.org) and the Transactions for society, as an archival journal, covering issues “transcending disciplinary boundaries.” The later establishment of Software Engineering Society as a technology area in SDPS completed the revolutionary developments in the redefinition of software engineering research to achieve transformative goals by transdisciplinary means. Dr. Dogru has been one of the founders of these initiatives and cut his teeth among the pioneers in the development of fundamental ideas and themes of modern advanced software engineering. Superb selection of topics for this book reflects many of these experiences and developments.

Now, more than forty years after the recognition of Software Engineering as a discipline, considering software as a glue; systems integration issues are again in the forefront of problems we are facing – from health care, biological/medical application, popular networking application, to defense applications. Maturity of technological infrastructures including support for Service Oriented Architecture (SOA) prepared the way to develop systems applications with integration being the paramount issue. Now, we are facing the problem to address industry-wide practical as well as theoretical issues on Systems Integration. Many researchers are developing solutions using integration tools in all kinds of areas from all engineering disciplines. However, there is no book in advanced software engineering to prepare the interested groups to these current fundamental issues and themes. This book serves this purpose. Obviously, no single book can cover extremely broad area of advanced software engineering. The goal is to prepare the minds so that the techniques for further study and usage will be clear. Nobody expects to learn Electrical Engineering by simply studying a book claiming to contain the Electrical Engineering body of knowledge. The broader and youthful nature of software engineering makes it even harder to teach in one comprehensive compendium. Therefore, the right approach has been taken by the authors to limit their coverage by taking into account the main themes and technologies, and covering a significant amount of advanced topics in the profession of software engineering.

We should remember that the challenges in software development are well documented and software engineers are struggling to offer increasingly more effective techniques. The prior serious attempts in this struggle seem to be targeting improvements in orders of magnitude in development time, while utilizing revolutionary techniques usually based on composition and architecture centered paradigms. The main difficulty in the process of software development, however, is not just the selection of any technical paradigms. Another key difficulty is the fusion of different disciplines: social aspects need to be mastered besides the formalism behind the problem model. In other words, multidisciplinarity is inherent in today’s software development. This book presents its content within this perspective, offering a coordinated mixture of different perspectives to address these frontiers. Practical techniques are also presented within this philosophy, adding to the value of the book in support of the multi-dimensional concern space. Get ready for the profession of the 21st century by studying the topics covered in this book.

Murat M. Tanik
October 2010, Birmingham, AL
Murat M. Tanik is a professor at the University of Alabama (UAB) at Birmingham since 1998. Prior to joining the UAB faculty, he was an associate professor and the director of Software Systems Engineering Institute (SSEI) at the University of Texas at Austin and served as the director of Electronic Enterprise Engineering at NJIT. He is also the director and chief scientist of Process Sciences Laboratory, a think-tank of process-centered knowledge integration. Dr. Tanik has worked on related projects for NASA, Arthur A. Collins (developer of Apollo moon missions’ tracking and communications systems), and ISSI. After Collins and ISSI, he joined SMU as an associate professor and the director of the Software Systems Engineering Technology (SEK) research group. Dr. Tanik is co-founder of the interdisciplinary and international society, Society for Design and Process Science. His publications include co-authoring six books, co-editing eight collected works, and more than 100 journal papers, conference papers, book chapters, and reports funded by various government agencies and corporations. Under his direction, more than 20 PhD dissertations and 25 M.S. theses have been completed. Dr. Tanik’s research interests include philosophy of science, software systems engineering, embedded and intelligent software systems, wireless and time-critical software support, collaborative computing for domain specific applications, and integrated systems design and process engineering. His first principles research include the development of information theoretical foundations for computing.