from the editor

IEEE Software—2007 and Beyond

Hakan Erdogmus

As I step into my new role as editor in chief and reflect on my predecessors’ achievements, the realization that I have pretty big shoes to fill is sinking in. The sentiment is a mix of anticipation and exhilaration. All right, I admit, throw some anxiety into the mix as well, but the healthy kind that comes over us when we’re about to start something really worthwhile. In my first column, I’d like to give you a glimpse of what to expect in the foreseeable future.

What IEEE Software is about

IEEE Software is in the business of building the community of leading and future software practitioners. It delivers timely, reliable, relevant, and leading-edge content to its target audiences. It also translates software theory into practice. Striving to strike the delicate balance between research and practice, it transfers ideas, methods, and experiences across academic and industrial communities. The magazine has a wide scope spanning multiple levels and perspectives of software development, from programming concepts to process considerations, from technical topics to managerial issues. We address the art, science, and business of software development in the blurry intersection of three intertwined realms: computer science (languages, programming paradigms, and foundations), software engineering (processes, methods, and tools), and software practice (technology, human factors, and economics). What also makes the magazine unique is its ability to bridge and unify these realms.

Who we write for

The magazine targets four distinct audiences. The first is the learner-practitioner—software professionals who want to remain on the leading edge. The learner-practitioner can be anyone participating in software development in any role, from programmers to executives and from small to large organizations. The second audience is the grounded inventor—professionals, consultants, thought leaders, and academics who create new concepts, theories, and techniques. The third is the grounded scholar—academics and researchers who diligently study various aspects of software development. The fourth audience comprises grounded educators who wish to maintain their curriculum’s relevance. The grounded inventor, scholar, and educator all have one common characteristic: they care about the learner-practitioner.

To satisfy our target audiences, IEEE Software’s Editorial Board, with a lot of input from the Advisory Board and help from the staff, solicits and screens contributions. Together, we engineer the content to be informative, horizon-expanding, accessible, impartial, novel, and sophisticated—all to ensure that readers can continue to count on IEEE Software as their primary source of new ideas, expert opinion, and practical advice.
Themes to look forward to

Alongside timeless, traditional software development topics, current and emerging trends will play a significant role in influencing the content. As Brian Fitzgerald and Michael Lang rightly pointed out in the March/April 2005 issue (“Hypermedia Systems Development Practices: A Survey”), the way we understand and carry out software development has been undergoing a transformation, if not a major paradigm shift. Many of us are revisiting the ivory-tower view of software development as a strictly engineering or purely process-driven discipline; we can observe this view converging with global trends outside the software industry and giving rise to a holistic approach. Some manifestations are integration across and widespread awareness of previously disparate development activities and levels, the dismantling of traditional silos, mixing and consolidation of heavy and lightweight methods, fluidity of roles, and recognition of human, social, and economic factors. The magazine will provide balanced, intelligent perspectives about these ongoing developments that affect our industry.

We’ll aim for more down-to-earth content, complementing the traditional process and management focus with technical topics. Thus, another important goal for the next few years will be to reorient the coverage toward issues fundamentally relevant to small and medium-size organizations as well as to individual developers. Revival of nuts-and-bolts topics that promote development of basic skills in coding, testing, design, modeling, and requirements is part of the plan. We’ll also cover budding, advancing, and changing software technology, including languages, frameworks, and environments. Some manifest trends include coverage of software engineering techniques for selected, red-hot domains. Finally, we’ll keep our eyes and ears open for opportunities in yet-uncharted territories: given the head-spinning pace of our industry, we’ll maintain sufficient flexibility to incorporate unforeseen topics into our calendar as they emerge.

New in this issue

To be able to cover the wide spectrum of topics planned, the Editorial Board will also enjoy an expansion. Incoming associate editors will supplement our pillar areas of requirements, design, quality, and management with new editorial areas. These include social and human factors; processes and practices; programming paradigms and languages; distributed and enterprise software; development infrastructures; and empirical results.
The IEEE Software Boards

My predecessor, Warren Harrison, takes on the position of editor in chief emeritus this month. Warren broadened the magazine’s scope, balanced its coverage, relentlessly promoted it in the community, increased its practical orientation, and recruited superb people. He also transitioned me into my new role. He deserves our deepest gratitude for his four years of invaluable service. Luckily, we aren’t saying farewell; we’ll continue to benefit from his vast editorial experience and professional insights.

Along with Warren and the Software staff, I also thank the other retiring members of the Editorial and Advisory Boards for years of dedicated support and outstanding contributions. Editorial Board members Jane Hayes and Nancy Eckelmann jointly produced the popular Quality Time column, and Wolfgang Strigel oversaw experience reports. Warren Keuffel is leaving the Editorial Board but will continue to look after Bookshelf, our book reviews department, as a contributing editor. Robert Cochran served on the Advisory Board both as chairperson and member. Also retiring from the Advisory Board are Melissa J. Murphy and Martyn Thomas. I wish all these colleagues success and fulfillment in their future endeavors.

Joining the Editorial Board are two new associate editors in chief, Frank Maurer and Helen Sharp, and columnist J.B. Rainsberger. Frank is a professor at the University of Calgary’s Computer Science Department. His expertise spans software process, agile methods, knowledge management, distributed software development, and Web-based systems engineering. Helen is a senior visiting research fellow at the City University Centre for Human-Computer Interaction Design, London, and a senior lecturer in the Open University’s Computing Department in Milton Keynes, UK. She’s an expert in qualitative research methods and user-centered approaches to software development, best known for her field studies of software practitioners and teams. Our new columnist J.B. Rainsberger, of Diaspar Software, is a renowned software development and testing expert, a teacher, a developer, and the author of the immensely popular JUnit Recipes.

There’s more. Joining the magazine’s Advisory Board this month are five new industrial members with diverse backgrounds, remarkable contributions, and deep community involvement: Frances Paulisch of Siemens, Dave Thomas of Bedarra Research Laboratories, Jennitta Andrea of ClearStream Consulting, our former Editorial Board member Wolfgang Strigel of QA Labs, and our outgoing editor in chief emeritus, Steve McConnell of Construx Software.

I welcome the new members to the IEEE Software family and look forward to working with them. —Hakan Erdoganmus

A new column, Not Just Coding, by J.B. Rainsberger debuts this month. J.B. maintains that most “programming” happens while the programmer isn’t coding. He refutes the notion that programming is “turning some smart person’s specification and design into code.” In Not Just Coding, J.B. encourages programmers to reach out beyond writing code to contribute in other ways to the success of their projects and teams. J.B.’s column will appeal to programmers of all levels. By addressing issues that fall outside the traditionally narrow perspective of programming, it will help developers who take pride in their work look at the array of issues and activities that make or break software projects, from conception to delivery and beyond.

Happy reading! ☺