Software Engineering — Components, Interfaces, Behaviours

Prof. Dr. Walter Dosch
Director, Institute of Software Technology and Programming Languages
University of Luebeck, Germany

Abstract

Software engineering has matured from heuristic practice to an engineering discipline. Over the years software technology developed into a key qualification for mastering complex technical systems. Nowadays software engineers can benefit from a solid stock of basic research addressing the specification, modelling, design and implementation of sequential, concurrent, distributed and real time systems. The talk surveys the scientific foundations of modern software technology concentrating on components, interfaces and behaviours. We present a unifying approach relating different system views manifesting themselves as data model, communication model, state transition model, and process model.

Biography

Dr. Walter Dosch received the Staatsexamen degree (equivalent to MSc) in Mathematics and Physics and the Dr. degree (equivalent to Ph.D.) in Computer Science from the Technical University of Munich in 1975 and 1987, respectively. From 1988 to 1996 he was Professor for Theoretical Computer Science and Artificial Intelligence at the University of Augsburg. Since 1997 he is full professor and Director of the Institute of Software Technology and Programming Languages at the University of Luebeck. During the election period 2000 to 2002, he served as Dean of the Faculty of Technology and Sciences at the University of Luebeck. Dr. Dosch joined the program committees of more than 60 international conferences and published over 130 articles in refereed proceedings and journals.

He served as the General Chair and Program (Co)Chair at conferences in the USA, Europe and Asia, and contributed invited plenary lectures to several conferences. He obtained major funds from the European Union under the ESPRIT and INTERREG programmes. His research interests focus on the foundations of software engineering including specification, modelling and design of sequential, concurrent and reactive systems. In the field of programming languages, the research concentrates on high-level programming concepts for parallel and distributed applications along with their efficient implementation.