Educating Students in Value-Based Design and Development

Barry Boehm
University of Southern California
boehm@sunset.usc.edu

Abstract

Much of software engineering is taught and practiced in a value-neutral context, in which every requirement, use case, object, test case, and defect is equally important. Too often, students learn that some of their stakeholders’ value considerations are more important than others by failing to consider this on the job and suffering the consequences.

The recent book, Value-Based Software Engineering (S. Biffl et al., eds., Springer, 2005) sets out the agenda of the value-based software engineering community. It is to integrate value considerations into traditional software engineering principles and practices for use in software engineers’ education and daily work.

We have been pursuing this agenda in a research project called “A Value-Based Science of Design,” within the NSF Science of Design program. This talk will address the nature of “value” in a software engineering context; present an initial theory and process for performing value-based software engineering; present example value-based techniques for requirements engineering, design, development, and test; and discuss experiences in incorporating value-based software engineering in individual-assignment and team project courses.

Dr. Barry Boehm is the TRW Professor of Software Engineering, Computer Science Department and the Director of the Center for Software Engineering at the University of Southern California (USC). Dr. Barry Boehm served within the U.S. Department of Defense (DoD) from 1989 to 1992 as director of the DARPA Information Science and Technology Office and as director of the DDR&E Software and Computer Technology Office. He worked at TRW from 1973 to 1989, culminating as chief scientist of the Defense Systems Group, and at the Rand Corporation from 1959 to 1973, culminating as head of the Information Sciences Department. He entered the software field at General Dynamics in 1955.

His current research interests involve recasting software engineering into a value-based framework, including processes, methods, and tools for value-based software definition, architecting, development, validation, and evolution. His contributions to the field include the Constructive Cost Model (COCOMO), the Spiral Model of the software process, and the Theory W (win-win) approach to software management and requirements determination. He has received the ACM Distinguished Research Award in Software Engineering and the IEEE Harlan Mills Award, and an honorary ScD in Computer Science from the University of Massachusetts. He is a Fellow of the primary professional societies in computing (ACM), aerospace (AIAA), electronics (IEEE), and systems engineering (INCOSE), and a member of the U.S. National Academy of Engineering.