The 29th IEEE International Conference on Software Engineering Education and Training

CSEE&T 2016

Dallas, Texas, April 5-6, 2016
http://paris.utdallas.edu/cseet16

Keynote Speech

Software Engineering Education in an Age of Specialization

Anthony J. Lattanze
Institute for Software Research
Carnegie Mellon University
http://mse.isri.cmu.edu/software-engineering/faculty-new/lattanze-anthony.html

Abstract

The MSE program was established in 1989 at Carnegie Mellon University with the aspiration of improving the state of the practice of software engineering. At 25 years, our program is one of the oldest Software Engineering Masters' degree programs in the world. Our programs are built around the concept of core courses, a project course, and electives. At the center of this structure is the philosophy that core courses impart the *fundamental enduring principles of software engineering* to our students. This is the "stuff" ALL software engineers should know and is the soul of our programs. However at the founding of the MSE program the World Wide Web of today did not exist. Embedded software and systems had not yet found a common place in household appliances, commercial electronics, and automobiles. Java did not exist. The smart phone had not been invented, and even the ordinary cell phone had not been widely adopted. There was no Facebook, Google, Amazon, or Yahoo. The state of the practice of software engineering was weak. Applications and systems were built around monolithic development models derived from traditional systems engineering methods. Software engineering generalists were desperately needed, and there was little need for the diverse specialization that we see in today's software industries. Today software engineers in different domains go about the process of designing, building, and deploying systems in radically different ways. Given this diversity, we are questioning the practicality of a single set of common software engineering core courses that can serve all software engineers and domains equally well. In this talk, I will present the challenges of adopting a single set of software engineering core courses and discuss some of the potential options we are exploring to better address the needs of tomorrow's specialized software engineers.

About the speaker

Anthony J. Lattanze is currently the director of the Masters of Software Engineering programs for the Institute for Software Research (ISR) at Carnegie Mellon University (CMU) and is one of the founders of the Embedded Software Engineering masters program. Anthony was also a member of the Software Engineering Institute's senior technical staff at CMU, where he was

instrumental in helping to develop the Architecture Tradeoff Analysis Method (ATAM) and the Quality Attribute Workshop (QAW). He led the development of the SEI's Architecture Training Program and helped to transition these methods into industries and organizations around the world. His primary research interest is in the area of software and systems design – especially as it applies to embedded, software intensive systems. He is the author of numerous articles, journal papers, textbook contributions, and is the author of "Architecting Software Intensive Systems: A Practitioners Handbook." In addition to his work at CMU, Anthony works with organizations around the world as an architecture and systems design consultant helping organizations with system design, providing design coaching, evaluating designs, and training. Prior to Carnegie Mellon University, Mr. Lattanze was the Chief of Software Engineering for the Technology Development Group at the United States Flight Test Center at Edwards Air Force Base, CA. During his 15 year tenure at the Flight Test Center, he was involved with numerous projects as a software engineer, software and systems architect, and project manager. Anthony was involved with development and test of aircraft such as the B-2 Stealth Bomber, F-22 Advanced Tactical Fighter, Air Borne Laser Test Bed, among other projects.