Abstract

This panel will discuss recent trends in graduate software engineering programs, including adoption of GSwE2009, cooperative programs between schools, increasing use of distance education formats, and specialization of programs for industry partners. Panelists will also discuss the evolving relationship of software engineering to other disciplines, such as computer science and systems engineering.

1. Position Statement and Biography of Mark Ardis

As moderator I will seek to engage the audience with the panelists, and vice versa.

Mark Ardis is a Distinguished Service Professor in the School of Systems and Enterprises at Stevens Institute, where he teaches courses in software engineering. He spent the first part of his computing career as a government contractor and later performed research on formal methods and product line engineering for Bell Laboratories. In his academic career Mark has helped establish software engineering programs at Wang Institute, Carnegie Mellon University, Rose-Hulman Institute of Technology, Rochester Institute of Technology and Stevens Institute of Technology.

2. Position Statement and Biography of Shawn Bohner

Rose-Hulman Institute of Technology’s Department of Computer Science and Software Engineering (CSSE) launched a new master’s degree in software engineering in the Fall of 2010. The degree builds upon our undergraduate software engineering program and incorporates the 2009 Graduate Software Engineering Reference Curriculum (GSwE) targeting largely working software professionals. Incorporating systems engineering into the curriculum provides a valuable edge for our students as the landscape of software systems increasingly entails interdisciplinary, model-based solutions rendered on hybrid computing platforms (e.g., traditional, parallel, mobile, reconfigurable). Further, Agile development
approaches coupled with the continued advancement in web and cloud technologies have shifted the emphasis in both software systems development and the technologies used to deliver the education.

Massive open online courses (MOOCs) are poised to change the landscape of higher education. Like financial shifts, tectonic shifts in longstanding institutions often take longer than anticipated, and are more disruptive than expected. The “elephant in the room” is whether this will substantially change software engineering education, and if so, how? Key MOOCs like Coursera, Udacity, and edX are well-financed providers and associated with top universities. These are starting to provide software engineering offerings since the marketplace demand is so high. With much of the software engineering curriculum taught via practicum (learn by doing), are things like the inverted classroom, and use of online distributed learning a potential advantage? Can existing institutions of high education weather the transition and even lead the transition to more effective teaching of engineering topics like software?

Shawn Bohner is a professor of computer science and software engineering at Rose-Hulman Institute of Technology. After a career in the software and systems engineering industry, he joined academe in 2001 focusing on software engineering. His areas of research include software engineering education, software maintenance and evolution, impact analysis, and dynamically reconfigurable computing systems. Dr. Bohner holds a PhD from George Mason University, and BS and MS degrees from University of Maryland and Johns Hopkins University respectively.

3. Position Statement and Biography of Dick Fairley

Several important events have occurred since publication of GSwE2009, including:

* revision of the SWEBOK 2004 knowledge areas (to appear in SWEBOK V3), and
* development of a framework for competency models plus work in progress on a competency model for software engineering practitioners.

This presentation will provide an overview of the changes to SWEBOK and the relevance of competency models for education and training.

Dr. Richard E. (Dick) Fairley is principal associate of Software and Systems Engineering Associates (S2EA) and an adjunct professor in the graduate software engineering program at Colorado Technical University in Colorado Springs, Colorado. He is chair of the IEEE Computer Society’s Software and Systems Engineering Committee. He is also chair of a joint Computer Society-Project Management Institute project to develop a software project management extension to the PMI Body of Knowledge (PMBOK®). In his career of 30+ years, Dr. Fairley has been a tenured professor, department chair, academic dean, and trainer and consultant. His research interest is in understanding and documenting the relationships between software engineering and systems engineering. He holds bachelors and masters degrees in electrical engineering and a PhD in computer science and applied math. Dr. Fairley is a member of IEEE, the IEEE Computer Society, INCOSE, and PMI.

4. Position Statement and Biography of Dennis Frailey

Like all new fields of study, software engineering has struggled to define itself over the past several decades. The discipline is slowly taking shape in the form of curriculum models, bodies of knowledge, accreditation criteria, certifications and even licensing; but we keep
learning new things that challenge the validity of these infrastructure elements as we build them. We must exercise ongoing care and diligence as we balance the influx of new methods and insights with the desire to define, codify and standardize what we do. At the same time, graduate programs in all disciplines have struggled to make the most appropriate use of emerging instructional technologies and delivery formats. Our advantage in software engineering is that our discipline underlies most of these emerging technologies and, hence, we have often served as pacesetters. The future holds more challenges as we seek to deliver affordable education to ever growing numbers of students around the world. We can and should continue to be leaders in the use of new instructional approaches and innovative delivery platforms. This will help validate our discipline in the eyes of others and accelerate our acceptance in the academic community and with the public.

Dennis Frailey recently completed a 36-year career as a computer and software engineer at Texas Instruments and Raytheon Corporation. At retirement he held the title of Principle Fellow. Although working for industry most of his life, Dennis has been deeply involved in the emergence of computer science and software engineering as academic disciplines and in providing high quality education in these fields. After graduating from Purdue University with a PhD in Computer Science (in 1971) he helped start both the Computer Science and, later, the Software Engineering program at Southern Methodist University; taught as a faculty member and, for most of his career, as an adjunct faculty member at SMU and the University of Texas at Austin; has served as a CSAB/ABET program evaluator since 1986; served on numerous ACM and IEEE-CS committees that helped establish software engineering as a profession; was chair of the Dallas Fort Worth Association for Software Engineering Excellence; helped develop GSwe2009; wrote part of the 2004 version of the SWEBOK Guide; and currently serves as the chair of the Computer Society Education Board's Professional Development Committee, which is providing high quality courses and other educational products to support professional software developers. For many years he's also been an ACM Distinguished Lecturer and IEEE Computer Society Distinguished Visitor, and has spoken on software and computer topics to over 150 chapters of these societies. Dennis has a close relationship with CSEET, having spoken at the conference on several occasions and served on the steering committee for many years. In 2008 he won the ACM SIGCSE's Lifetime Achievement Award in computer science education and this year he is the recipient of the Nancy Mead award for outstanding contributions to software engineering education.

5. Position Statement and Biography of Tom Hilburn

Many, if not most, graduate software engineering programs (at least at the master’s level) have a principal goal of preparing graduates for professional software engineering practice. For example, the Graduate Software Engineering 2009(GSwE2009) guidelines are intended to support professional practice, and a review of the GSwE2009 student outcomes show an emphasis on professional knowledge and practice.

It is important for professionally-oriented programs to establish some relationship between the design and delivery of their programs and the needs of software development organizations that employ their graduates. During the development of GSwE2009 a survey and study of 28 programs assessed the professional content of those programs, using the SWEBOK as a software engineering competency model.

Recent work on competency models for software engineering, systems engineering and software assurance offer ideas for how to develop or improve a program, which has a goal of preparing students for professional practice. These models provide a view of how
professional practice is categorized and how a professional career might be structured. Such a view provides the context and framework for determining program goals, and the type and level of software engineering practice that is appropriate and desirable for inclusion on a graduate software engineering program.

Dr. Thomas B. Hilburn is a Professor Emeritus of Software Engineering at Embry-Riddle Aeronautical University and was a Visiting Scientist at the Software Engineering Institute, Carnegie-Mellon from 1997 – 2009. He has worked on software engineering development, research, and education projects with the FAA, General Electric, Lockheed-Martin, the Harris Corp, the MITRE Corporation, DOD, FIPSE, the SEI, the NSF, the ACM and the IEEE Computer Society. He is an IEEE Certified Software Developer, an IEEE Software Engineering Certified Instructor, and has chaired committees on the Professional Activities Board and the Educational Activities Board of the IEEE Computer Society.

6. Position Statement and Biography of Greg Hislop

Drexel offers both BS and MS degrees in Software Engineering. The MSSE, like many computing Masters degrees, primarily attracts students interested professional practice rather than research. Key issues in the evolution of this program include:

Admission Requirements – The pool of BSSE graduates is still relatively modest, and the MSSE must provide a reasonable entry path for students with undergraduate majors other than software engineering. Defining those requirements is not a simple task. While other computing majors would seem like a close fit, some computing graduates have little appreciation for large scale software systems and other concepts fundamental to software engineering. Related areas such as traditional engineering disciplines may provide even less preparation. Industry experience can be excellent preparation, but is often difficult to evaluate based on admissions materials.

Online Education – Online education is clearly growing, but level of penetration is very uneven across higher education. Professionally oriented Masters degrees are one segment where online education can be fruitfully applied, and Drexel has significant experience with this delivery mode. Even for developed programs however, faculty development, and faculty self-concept raise questions about how to ensure quality program delivery over time.

Gregory W. Hislop is on the faculty of the iSchool at Drexel University where he has worked extensively on curricula for Software Engineering, Information Systems, and Information Technology degree programs. Prior to that he spent about 20 year as a computing professional. His current work includes exploring ways for students to learn computing by participating in humanitarian free and open source software projects.

7. Position Statement and Biography of Todd Sedano

Often industry feels a disconnect between what is taught in the classroom and the real world environment. We see a shift away from traditional educational approaches. Carnegie Mellon University in Silicon Valley's tagline, "Developing Software Leaders" captures the essence of our educational philosophy. Our students need to master modern software engineering and management methods, learn to align project decisions with business goals, and develop the communication and teamwork skills.

The students achieve these skills through authentic project work reflective of real-world scenarios. We use "learn-by-doing" and "story-centered curriculum" pedagogy as the
mechanism for the students to acquire these skills. We minimize the transfer between the educational experience and the workplace by situating the student in a realistic, problem-centric context. When we can, we have students experience the learning objectives by executing a real-world or synthetic project. If implementing a project is too time consuming, then we simulate the learning. Thus, our students create authentic work products in a team setting, which is exactly the way they will be required to perform in industry.

Todd Sedano is the Director of the Software Engineering Program for Carnegie Mellon University's Silicon Valley campus. In addition to managing the day-to-day operations of the Software Engineering program, he teaches the craft of software development, agile methodologies, entrepreneurship, and improv to his graduate students. He runs "Improv for Engineers" workshops that allow engineers to develop skills in public speaking, active listening, idea building, confidence, and team formation.

While working in the Artificial Intelligence group at the Jet Propulsion Laboratory, Todd worked on planning and scheduling software for the Deep Space Network antennas and space probes, image processing, and system testing of the Remote Experiment Agent planning software on the Deep Space 1 spacecraft. In Silicon Valley, Todd has worked with both startup and established companies. Using J2EE technologies he has worked on websites for Sega, Roxio, Target, Better Homes and Gardens, and CustomerNation; integrated software with FedEx, UPS, and NetSuite webservices; been in a few death marches; and managed several development teams.

Developing on the Commodore PET was Todd's first paid programming experience.