Designing Software Engineering Option within Computer Science Bachelor of Science Program

Jan Bergandy, Paul Bergstein
Computer and Information Science Department
University of Massachusetts Dartmouth
Dartmouth, MA 02747, USA

Boleslaw Mikolajczak, Haiping Xu
Computer and Information Science Department
University of Massachusetts Dartmouth
Dartmouth, MA 02747, USA

Abstract *- This paper describes a Software Engineering Option (SEO) within the current BS in Computer Science program at UMASS Dartmouth. The SE Option reuses significant portion of the existing Computer Science curriculum and provides exposure to software engineering discipline through five new courses and a yearlong senior group software engineering project. This approach is innovative because it builds SE Option on solid foundations of ABET-accredited object-oriented computer science curriculum, it incorporates a capstone year-long senior group software engineering project that involves computer industry as projects' sponsor and consumer, it builds students' competence over a period of three years, it provides economical alternative to a full-blown Software Engineering program, it assures ABET-accreditability of the SE Option within BS in Computer Science program, it allows students from the Community College Computer Science Transfer program to join the SE Option.*

Keywords: Software Engineering Option, ABET-accredited Computer Science Program.

1 Introduction

The Computer and Information Science (CIS) Department offers BS in Computer Science since 1983 and MS in Computer Science since 1987. Since 2005 we also offer a BS/MS in Computer Science. The BS in Computer Science program was accredited in 1988 as the first accredited Computer Science program in the state of Massachusetts. The program was continuously accredited since then. The current accrediting agency is ABET. During these years we graduated circa 800 computer science majors, 500 of them with BS in Computer Science degree and 300 with MS in Computer Science degree. The CIS Department has 14 full-time tenured/tenure-track faculties.

The ability to mount the SEO depends on three components: faculty, infrastructure and students.

Faculty: There are faculty with specialties in software maintenance and evolution, object-oriented software technologies, software engineering of parallel/distributed systems, software testing and verification, applications of formal methods in software design, design of parallel and distributed software systems, and software engineering education. Faculty has taught six undergraduate courses directly related to the SE Option. These six courses are being reused as foundational courses. Five additional required software engineering courses are being developed plus yearlong senior software engineering project.

Infrastructure: There exist specialized labs that fall under the SEO. We maintain state-of-the-art software labs in Linux and Windows environments. The Advanced Technology Manufacturing Center (ATMC) at UMASS Dartmouth serves as resource for the realization of cooperative learning and internship programs.

Students: The undergraduate program in computer science is accessible to high school graduates, to transfer students from community colleges, and to employees in regional industries. The CIS Department has had an annual population of 200-250 undergraduate computer science majors and 100 graduate computer science majors during the 2000-2006 academic years.

2 Need for Software Engineering

The Commonwealth of Massachusetts is undergoing a rapid change in the nature, diversity, and distribution of their industries. The traditional energy-intensive industries are giving way to a variety of information-intensive industries which rely on software, computer systems, and their supporting infrastructures and applications. Recent studies by the US Department of Labor, the Massachusetts Technology Collaborative, the New England Journal of Higher Education and Economic Development, and the Massachusetts Technology Leadership Council highlight a growing need for highly qualified software engineers having scholastic levels above the associate degree to

handle the problems found within these emerging industries. Specifically:

- According to the 2005-06 edition of *The Complete Guide to the Massachusetts Software Industry*, published by the Massachusetts Technology Leadership Council [1], there are in Massachusetts 2,655 software-oriented companies. These 2,655 software companies currently produce annual revenue totaling $10.5 billion and employ 118,976 employees.
- The US Bureau of Labor Statistics, *Occupational Outlook Handbook* [2] states "Computer software engineers are projected to be one of the fastest-growing occupations from 2004 to 2014. Employment of computer software engineers is expected to increase much faster than the average for all occupations, as businesses and other organizations adopt and integrate new technologies and seek to maximize the efficiency of their computer systems."
- Also according to the BLS Occupational Outlook Handbook "jobs in software engineering are less prone to being sent abroad compared with jobs in other computer specialties, because the occupation requires innovation and intense research and development."
- According to Massachusetts Technology Leadership Council President Joyce L. Plotkin, "The state's software industry has grown threefold since the first Mass Software Council survey in 1989…"
- The Massachusetts Board of Higher Education's Computer and Information Systems Technology (CIST) report [3] emphasized the need for advanced degrees in computing and in software engineering in particular.
- In April 2006, Money Magazine and Salary.com [4] rated Software Engineer as the No. 1 Best Job in America, with 44,800 annual job openings, a projected 10-year growth of 46%, and an average salary of $80,500.

The evidence is compelling that investing in long range research in computing is extremely profitable for society. In particular, investing in software engineering education serves extremely well the state and regional interests.

"*Since a major function of the Software Engineering degree is to focus on applications of computer science theories and methodologies to development of correct and reliable market product called software, students who are not exposed to these ideas may be handicapped in the workplace*" [5].

The Software Engineering Option complements existing IT programs while filling this gap for students at UMASS Dartmouth, students in community and state colleges across Southeastern Massachusetts and beyond. The option is the first in the state of Massachusetts and also the first in the New England macro-region.

Computer Science, Computer Engineering, Software Engineering, Information Systems, and Information Technology are now recognized as independent academic disciplines. Therefore, to adequately compete in the marketplace, Software Engineering Option is a necessity. Taking into account predicted growth of the job market for software engineers we expect solid enrollment figures for Software Engineering Option [2, 4]. There are currently circa 10 ABET accredited B.S. programs in Software Engineering nationally. In order for the state of Massachusetts to maintain its position at the forefront of engineering education, it must address this key emerging area of engineering. The Software Engineering Option contributes to this objective.

The Computer and Information Science (CIS) Department offers BS in Computer Science since 1983 and MS in Computer Science since 1988. The BS in Computer Science Program was accredited in 1988 by CSAB as the first accredited Computer Science program in the state of Massachusetts. The program was continuously accredited since then. The current accrediting agency is ABET. During these years we graduated 800 computer science majors, 500 of them with BS in Computer Science degree and 300 with MS in Computer Science degree. The CIS Department has currently 14 full-time tenured/tenure-track faculty members. Six of them are highly qualified to provide instructions in software engineering discipline. Therefore the CIS Department is uniquely and highly qualified to offer a SEO within accredited BS in Computer Science program. The ability to mount the SEO will depend on three components: faculty, infrastructure and students.

Faculty: The SEO will be based on the strengths of the existing undergraduate program in Computer Science. There are six faculties with specialties involving software engineering such as software maintenance and evolution, object-oriented software technologies, software engineering of parallel and distributed systems, software testing and verification, applications of formal methods in software design, design of parallel and distributed software systems, and software engineering education. Faculty over the past few years, have taught six undergraduate courses directly related to the SE option. These six courses will be reused in the SEO as foundation courses. Five additional software engineering courses need to be developed plus yearlong senior software engineering project.

Infrastructure: There already exist recently developed specialized labs that would fall under the SE Option. We also maintain state-of-the-art software labs in Linux and Windows environments. The Advanced Technology Manufacturing Center (ATMC) at UMASS Dartmouth will serve as another resource that is already in place for the realization of cooperative learning and internship programs through academic as well as industrial grants and contracts. The goal of the ATMC is to provide advanced technology and manufacturing solutions, through industry and university partnerships, to meet business needs.

Students: The undergraduate program in computer science is accessible to high school graduates, to transfer students from community colleges, and to employees in regional industries seeking new career and employment opportunities. The CIS Department has had an annual population of 200-250 undergraduate computer science majors and 100 graduate computer science majors during the 2000-2006 academic years.

There are currently circa 20 BS in Software Engineering programs in the US, 10 of them are accredited by ABET. In June 2006 we conducted a survey among chairs of departments that house computer science and software engineering programs. It was evident that opening SE program brings new students into these programs. The SE Option is replicable to any BS in Computer Science program that satisfies the following generic conditions:

a) has a significant object-oriented component in the undergraduate curriculum

b) has a similar set of required computer science courses

3 Goals of the SE Option

The goals of the Software Engineering Option (SE Option) in terms of measurable outcomes are:

a) to develop SE Option within BS in Computer Science program at UMASS Dartmouth

b) to develop course materials in electronic form for five new software engineering courses and for year-long group software engineering project

c) to strengthen collaborations with IT companies in Massachusetts by using industry sponsored projects for the year long senior software engineering project

d) to develop advising materials (both course and career advising) for SE Option in a form of Software Engineering Student Handbook

e) to start implementation of the SE Option beginning in fall 2007 with admission of entering computer science freshmen.

The focus of the SE Option will have emphasis in the following areas: 1) the application of formal methods in design and development of software; 2) application software 3) systems software 4) software maintenance and software evolution 6) software process and software quality. The educational objectives of the program include:

- Graduates showing proficiency in the software engineering discipline. Graduates with acquired state-of-the-art knowledge and ability to apply this knowledge in practicing their profession as a software engineer.
- Graduates who are able to formulate, analyze, and solve software engineering problems taking into account technical, economical, and social constraints.
- Graduates who in their professional career can independently pursue the continuous learning process and mastering of their skills.
- Graduates who are equally prepared to function in their profession as an individual and as a member of a team.
- Graduates who can effectively communicate orally and in writing with wide range of audiences including all stakeholders of the software engineering process.

The management goals of the SEO include:

- Attracting highly qualified applicants
- Having a new enrollment of 15-20 full-time and 5-10 new part-time students per year
- Reaching a program steady state of 20-30 full-time and 10-15 part-time students
- Becoming a recognized "center of excellence" contributing to software engineering education, state-of-the-art research, knowledge acquisition and the economic growth of the region in software methodologies/technologies; and
- Creating a synergy of university-industry-commerce collaboration in research and in educating highly qualified software engineering workforce in the state of Massachusetts.

3. 1. Innovative nature of the SE Option

The innovative nature of the SE Option stems from several factors:

a) it builds a SE Option on solid foundations of ABET-accredited computer science program

b) it incorporates a capstone year-long senior group software engineering project that involves industry as projects' sponsors and consumers

c) it builds students' competence as software engineers over a period of three years (sophomore, junior, senior years)

d) it provides economical alternative to a full-blown BS in Software Engineering program

e) it assures ABET-accreditability of the SE Option within BS in Computer Science program

f) it allows students graduating from Community College Computer Science Transfer program to join SE Option.

UMASS Dartmouth offers currently the following IT-related degree programs: MS in Computer Science and MS in Computer Engineering, BS in Computer Science and BS in Computer Engineering, BS in Management Information Systems, and BS in Computer-Oriented Mathematics. UMASS Dartmouth offers numerous IT courses through various academic departments that will lead to a planned IT Minor/IT Certificate at UMASS Dartmouth. The CIS Department takes a lead in these efforts. There are also considerations to develop MS in Software Engineering program.

3.2. Broader impacts

The SE Option will be accessible to Computer Science Transfer students from community colleges (CC). Students attending CC are typically coming from first generation

immigrants as well as economically disadvantaged families. As a result the program will support broadening participation in IT education and workforce. The program will provide the Massachusetts IT industry with highly qualified workforce prepared to meet the challenges of software engineering profession. There is no other software engineering program offered at this point in the State. SE Option provides a template for other institutions to build a similar SE option/program. After successful implementation of the SE Option a transition to a separate BS in Software Engineering program shall be seamless.

We plan a direct involvement of industry in the SE Option. Currently the CIS Department has a Computer Science Industrial Advisory Committee (CSIAC) that meets twice a year to consider quality of educational programs of the department. The members of CSIAC expressed enthusiastic support of the SEO during meetings. Members of the board are coming from Raytheon, General Dynamics, Naval Underwater Warfare Center, Fidelity Investments, and other computer-related companies. We will engage board members as customers generating capstone software engineering projects and as industrial supervisors of these projects. The Spring CSIAC meeting every year will be devoted to presentations of final versions of student projects. CSIAC members and private sector will be also involved in internships and COOP positions for students enrolled in the SE Option. College of Engineering and CIS Department already have well-developed COOP and internship programs.

4 Rationale

This rationale has been influenced and is partly based on the following report "Software Engineering 2004, Curriculum Guidelines for Undergraduate, Degree Programs in Software Engineering", a volume of the Computing Curricula Series Aug. 23, 2004 prepared by IEEE/ACM Joint Task Force. In particular text in italics represents quotes from this report.

4.1. Importance of Software Engineering

Software plays a central role in almost all aspects of daily life. The number, size, and application domains of computer programs have grown dramatically; as a result, hundreds of billions are being spent on software development, and the livelihood and lives of most people depend on the effectiveness of this development. Software products have helped us to be more efficient and productive. They make us more effective problem solvers, and they provide us with an environment for work and play that is often safer, more flexible, and less confining.

Software engineering emerged during last 35 years as younger "sibling" of computer science and as a separate discipline of study and practice. It is apparent that costs of software development are not diminishing with progress of time. This indicates that software science and methodologies of software development and deployment have not yet been fully understood and/or applied in practice to achieve desired cost benefits.

There are serious problems in the cost, timeliness, and quality of many software products. The reasons for these problems are many and include the following*:*

- *Software products are among the most complex of man-made systems, and software by its very nature has intrinsic, essential properties (e.g., complexity, invisibility, and changeability) that are not easily addressed [Brooks 95].*
- *Programming techniques and processes that worked effectively for an individual or a small team to develop modest-sized programs do not scale-up well to the development of large, complex systems (i.e., systems with millions of lines of code, requiring years of work, by hundreds of software developers).*
- *The pace of change in computer and software technology drives the demand for new and evolved software products. This situation has created customer expectations and competitive forces that strain our ability to produce quality of software within acceptable development schedules.*

4.2. Definitions of Software Engineering

Over the years, numerous definitions of the discipline of Software Engineering have been presented. We highlight two definitions:

- *"Software engineering is that form of engineering that applies the principles of computer science and mathematics to achieving cost-effective solutions to software problems." [CMU/SEI-90-TR-003]*
- *"The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software" [IEEE 1990].*

One particularly important aspect is that software engineering builds on computer science and mathematics. These definitions clearly state that software engineering is about creating high-quality software in a systematic, controlled, and efficient manner. Consequently, there are important emphases on analysis and evaluation, specification, design, and evolution of software. In addition, there are issues related to management and quality, to novelty and creativity, to standards, to individual skills, and to teamwork and professional practice that play a vital role in software engineering.

4.3. Software Engineering as a Computing Discipline

As the size, complexity, and critical importance of software grew, so did the need to ensure that software performs as intended. By the early 1970's, it was apparent that proper software development practices required more than just the underlying principles of computer science; they need both the analytical and descriptive tools developed within computer science and the rigor that the

engineering disciplines bring to the reliability and trustworthiness of the artifacts they engineer.

Software engineering thus is different in character from other engineering disciplines, due to both the intangible nature of software and to the discrete nature of software operation. It seeks to integrate the principles of mathematics and computer science with the engineering practices developed to produce tangible, physical artifacts.

Drawing on computing and mathematics as foundations, software engineering seeks to develop systematic models and reliable techniques for producing high-quality software; and these concerns extend all the way from theory and principles to the development practices that are most visible to those outside of the discipline. The definition of the body of the Software Engineering Education Knowledge (SEEK) reflects the reliance of software engineering on computer science, with the largest component of the SEEK being Computing Essentials.

5 Software Engineering Option at UMASS Dartmouth

The option is consistent with the mission of the university that considers contribution to economical development of the Southeastern Massachusetts region as essential. The option is also consistent with the mission of the College of Engineering because it builds on the strength of existing ABET-accredited computer science program and the needs of industry. SE Option at UMASS Dartmouth [6] will promote and disseminate software science and methodologies of software development and deployment to computer industry. This should improve quality of software development and should lead to cost reduction of software development. Software engineering inherits (content-wise) significantly from computer science: the fundamentals, formalisms, structural and behavioral modeling, methodologies, and applies them to design of marketable product called software. Software engineering deals with systematic development and deployment of software systems that are significant in size and that are mission critical.

Software engineering involves architectural design of software systems, their structural design, and implementation on specific programming platforms. Software engineers optimize architectural and design decisions that support many practically important quality attributes such as reusability, modifiability, scalability, compositionality, performance, reliability, availability, etc. Software engineers operate in various application domains. Therefore understanding of software requirements in these domains is critical as well as is an ability to formally specify related requirements. Software engineers work in industry as application software engineers and as systems software engineers. They work as software quality assurance specialists, as designers and implementers of complex graphical user interfaces, as software usability specialists, software maintenance engineers, and software process specialists involved in project management of software product lines.

5.1. Required Courses of the Software Engineering Option

We envision the SE Option as strategic enhancement of existing BS in Computer Science program. The following required courses constitute core of the SE Option and the program's dependency graph can be found at [6]:
CIS 264 Software Quality Assurance and Testing – sophomore level course
CIS 290 Software Architectures and Frameworks– sophomore level course
CIS 365 Software Process and Process Management – junior level course
CIS 390 Design of Large Software Systems– *junior level course*
CIS 461 Formal Methods for Software Engineering – senior level course.

5.2. Software Engineering Project I & II

The SE Option provides students with an opportunity to develop a wide range of skills important to any software engineer. These skills will be backed-up by solid knowledge of concepts, facts, and methodologies. Students will develop these skills hands-on in laboratories associated with many courses and through challenging projects (including group projects). The faculty members of the SE Option recognize the need for some form of culminating experience where all these acquired skills could be tested giving all stakeholders confidence that students are ready to take real jobs in software industry.

Software Engineering Project as a Capstone Project meets the following objectives:

- provides an environment for integration of knowledge and skills acquired through semesters of study of software engineering
- creates an environment stimulating the reflection on the value of previously acquired knowledge and skills
- exposes students to the reality of the working environment in industry
- provides students with an experience with a large-scale problem from conception to implementation of the solution
- allows students to experience the need for software process (need for organization) as a function of project's scale and criticality
- teaches the balance between individual and group responsibilities
- exposes students to an environment where an individual answers to more than one entity (customer, project-lead, team) including an entity external to the organization
- demands polishing the presentation and negotiation skills acquired in previous courses

- asks for reflection on the role of compromise in context of interpersonal interactions as well as in the context of technical choices and time management
- forces the students to reflect on the role of context on selection of specific methodology or tool in problem solving
- develops in students the ability to adapt to new and unexpected situations encountered in a typical working environment

5.3. Challenges of Capstone Projects

We have identified the following key issues of implementing the SE Capstone Project (CP): identifying stakeholders and defining their roles, selecting a process to be used in conducting each project, designing a timeline for the project, examining the tools available versus needed, defining deliverables, taking inventory of skills of a typical student, assessing student's ability "to learn on the job" (project), deciding on a strategy for forming student teams, deciding on the process of matching the teams with projects. Projects will be solicited from local companies and from University entities. Project sponsor/customer will be responsible for defining the problem to be solved, negotiating the scope, providing a contact person, and providing a subject-matter expert. The customer will participate in requirements modeling process, provide testing environment for the product, and take ownership of the product. Instructor in charge of the CP will serve as a project manager and will play a leading role in project improvement process.

5.4. Choosing a Software Process for Capstone Project

One of the key factors in selecting the most suitable software process is the level of process formalism. The SE Capstone Project will be best served by using a low-to-medium formal process. The Capstone Project, due to its scope and duration, will involve projects developed by a single small team for one customer. We chose Rational Unified Process (RUP) for the following reasons:

- it is a good match for the project with a group of students over the period of two semesters
- it is use case driven – students are familiar with use cases through other courses
- it is strongly biased to development of systems based on object paradigm
- it promotes component-based development – students are familiar with this approach
- it emphasizes risk mitigation early and through the entire process
- it promotes defining and base lining the architecture early
- it is simple, straightforward, iterative, well supported
- it is configurable – RUP Framework supports process configuration.

5.5. Impact on Program Accreditation

In order to insure that SE Option would not adversely affect the ABET accreditation of our Computer Science program we have analyzed the areas where the SE Option could impact the accreditation criteria. The two areas are *Curriculum* and *Objectives and Assessments.*

Curriculum : Compared to the current program requirements, the requirements for SE Option differ as follows:

1. The following core courses will not be required for students in the SE Option: CIS 273 Computer Organization and Design, CIS 360 Algorithms and Data Structures, CIS 361 Models of Computation, CIS 370 Operating Systems Design, and CIS 481 Parallel and Distributed Software Systems .
2. The following courses are added to the required core for students in the SE Option: CIS 264 Software Quality Assurance and Testing, CIS 290 Software Architectures and Frameworks, CIS 365 Software Process and Project Management, CIS 390 Design of Large Software Systems, and CIS 461 Formal Methods in Software Engineering.
3. One of the technical elective courses, CIS 431 Human Computer Interaction, becomes a requirement in the SE Option, and the number of technical electives is reduced from 4 to 3. Also, the set of courses that can be counted as technical electives in the SE Option is more restrictive compared to the Computer Science technical electives.

The total number of hours in core and advanced Computer Science courses is unchanged in the SE Option. The requirements in all other areas (Mathematics, Science, and Humanities) are identical to the existing requirements for CS students. Since some of the CS requirements are changed for students in the SE Option, we verified that all ABET CS curriculum requirements would be met by the SE Option. In particular, we have verified the ABET requirements that the curriculum:

- Provides basic coverage of Algorithms, Data Structures, Software Design, Concepts of Programming Languages, and Computer Organization and Architecture.
- Stresses theoretical foundations, problem analysis, and solution design within the program's core requirements.

5.6. Objectives and Assessments

The SE Option is fully consistent with current Program Outcomes. The following Table 1 summarizes program's outcomes and where each is addressed in the SEO curriculum in terms of required courses.

6 References

[1] Mass Technology Leadership Council, Annual Report, *2005: Connecting Technology and Leadership*, http://www.masstlc.org/abo/reports/documents/3_MTLC_AR_2005.pdf

[2] United States Department of Labor, Bureau of Labor Statistics, Occupational Outlook Handbook, 2006-07 Edition, http://www.bls.gov/oco/ocos267.htm

[3] Mass Board of Higher Education, Information Technology Report, September 1999.

[4] Best Jobs in America http://money.cnn.com/magazines/moneymag/bestjobs/

[5] *Software Engineering 2004, Curriculum Guidelines for Undergraduate Degrees in Software Engineering, A Volume of the Computing Curricula Series*, The Joint Task Force on Computing Curricula, IEEE Computer Society and Association for Computing Machinery, Aug. 23, 2004

[6] http://www.umassd.edu/engineering/cis/courses/swewelcome.cfm

Outcome	Links to Curriculum
1. Solving problems algorithmically	Throughout entire curriculum
2. Applying knowledge of Mathematics and Science	CIS 290 Software Architectures and Frameworks CIS 390 Design of Large Software Systems CIS 461, Formal Methods in Software Engineering CIS 498, Software Engineering Project I CIS 499 Software Engineering Project II
3. Applying analytic and empirical techniques	CIS 264 Software Quality Assurance and Testing CIS 362 Empirical Methods for Computer Science
4. Designing system, component, or process	CIS 280 Software Specification and Design CIS 290 Software Architectures and Frameworks CIS 390 Design of Large Software Systems
5. Participating in multidisciplinary teams	CIS 498 Software Engineering Project I CIS 499 Software Engineering Project II
6. Solving problems involving information technology	CIS 365 Software Process and Project Management CIS 381 Social and Ethical Aspects of Computing
7. Articulating social, professional, ethical, and legal aspects	CIS 381 Social and Ethical Aspects of Computing
8. Evaluating impact at global/societal level	CIS 365 Software Process and Project Management CIS 381 Social and Ethical Aspects of Computing
9. Analyzing contemporary issues	Throughout entire curriculum
10. Communicating effectively	ENL 266 Technical Communication CIS 365 Software Process and Project Management CIS 498 Software Engineering Project I CIS 499 Software Engineering Project II
11. Applying modern skills, techniques, and tools	CIS 280 Software Specification and Design CIS 264 Software Quality Assurance and Testing CIS 290 Software Architectures and Frameworks CIS 365 Software Process and Project Management CIS 390 Design of Large Software Systems

Table 1. Mapping of program's outcomes onto required software engineering courses