Internationalization of Computer Science Education

Sarah Douglas, Art Farley, Ginnie Lo, Andrzej Proskurowski, Michal Young
Department of Computer and Information Science
University of Oregon, Eugene, OR, USA
[douglas, art, lo, andrzej, michal]@cs.uoregon.edu

ABSTRACT

Internationalization of computer science education involves incorporating awareness, knowledge and skills of professional life in a global environment. Through an NSF CPATH grant[1] we have established a Pacific Rim community of computer science departments, high tech industry and international programs exploring a new model of computer science education that focuses on the knowledge, skills and competencies necessary for professional success and leadership in a global context. This paper describes our progress in building an international community of computer science educators, as well as our efforts in curricular innovation and establishment of international summer schools. Internationalization of computer science education will help attract the best and brightest students and broaden the appeal of computer science to a much more diverse population. Computer science will be seen as a pathway to a career not in an isolated cubicle but in the wide-open world.

Categories and Subject Descriptors

K.3.2 [**Computers and Education: Computer science education, curriculum**]:

Keywords

Computer science education, internationalization, computers in society

1. INTRODUCTION

Today, both scientific inquiry and economic enterprise are global, but computer science education has remained largely parochial in both course content and student experiences. This raises the following important question: *What must we do to transform American undergraduate computer science education to prepare our students to work and to lead in a global community of computing professionals?* We will call this transformation internationalization, or *i18n*, borrowing the abbreviation for "internationalization" from the software development community.

Through a series of workshops and other activities we have established a Pacific Rim community of computer science departments, high tech industry and international programs exploring a new model of computer science education that focuses on the knowledge, skills and competencies necessary for professional success and leadership in a global context. Our community and experience can serve as a resource and model for creating similar communities nationwide. In this paper, we report on the activities of our community and discuss the results of our international collaboration toward building a common understanding of the importance of internationalization for computer science education.

2. COMMUNITY BUILDING

An integral part of our effort to develop an international element to computer science education has been to create a community dedicated to exploring the issues involved. Given our location in the Pacific Northwest, we focused on the Pacific Rim as a domain for creating such a community. Here we review the activities undertaken to create our community and develop ideas regarding internationalization.

2.1 Planning Workshop

In November, 2007 we conducted a planning workshop involving over twenty Pacific Northwest[2] academic and industry partners to refine our plans for community building. We developed a website, http://www.cpathi18n.org, to serve as a clearinghouse for information about our project. We also designed and printed a promotional brochure. During the planning meeting in November 2007 we identified areas of high priority to our Pacific Northwest academic and industrial partners, including a need for 'soft skills' in the areas of cross-cultural communication, team work across international and time-zone boundaries, the importance of face-to-face meetings for global team members,and the lack of international experience for American CS undergraduates.

2.2 Friendship Tour

To reach out to possible Asian partners and form initial connections, in March 2008, we visited several potential academic and industrial partners in China and Japan, to begin

[1] NSF grant CNS-0722341: CPATH CB: *i18n*, Internationalization of Computer Science Education: The Pacific Rim Community Model

[2] *North American PNW*; this US-centric error was pointed out by our Asian partner!

SIGCSE'10, March 10–13, 2010, Milwaukee, Wisconsin, USA.
Copyright 2010 ACM 978-1-60558-885-8/10/03 ...$10.00.

initial discussions of issues of mutual interest and to extend a personal invitation to the upcoming May workshop. The visits included presentations from the *i18n* team and hosts, as well as discussions of the perceived problems and possible resolutions. The institutions visited were Peking University, Tsinghua University, Beihang University, IBM China, Microsoft Research Asia, Kyoto University and Tohoku University. During our Asian visit we discovered that the issues facing us and our students are very similar to those facing our Asian partners. This indicated a strong possibility of developing mutually beneficial activities and projects.

2.3 International Workshops

We conducted three workshops over a one year period to build our Pacific Rim Community and address issues of internationalization of computer science education.

Workshop 1: Assessing the Needs and Challenges of i18n of Computer Science Education

The May 2008 workshop took place in Portland, Oregon. It represented the official launching of the project. Fifty participants from North American Pacific Northwest and Asia, including representatives of the computing industry, were introduced and the i18n management team members presented the project overview. Keynote addresses by Lionel Ni, chair of computer science department at HKUST and Gurdeep Singh Pall, corporate VP of Microsoft, introduced us to cross-cultural issues in academia and industry. The first session had as its theme 'Internationalization: Skills and Challenges'. Small working groups discussed and recorded their candidates for skills needed to perform effectively in a global context and impediments to acquisition of these skills.

The second session was focused on 'International Experiences'. The topics and generated ideas included industrial internships, student and faculty exchanges, a summer school, international volunteer service, sister departments and possible dual degree programs. The third session was dedicated to curricular innovations. Working groups topics included: an i18n model curriculum, international software engineering projects, international software engineering competition, a cross-cultural communication course, a cultural HCI curriculum module and an international ethics curriculum module. We found a strong desire for interaction across the Pacific Ocean and an eagerness for educational collaboration.

Workshop 2: Enhancing Computer Science Education through Pacific Rim Partnerships and International Experiences

Working groups that formed during the May 2008 workshop reported their findings at the next workshop, which was held in Beijing during October 2008 on the campus of Peking University, the workshop co-sponsor. The workshop was attended by over sixty participants that included deans, department heads, professors, and students from our Pacific Rim partner institutions in the U.S., Canada, China, Japan, Korea, the Philippines, and Vietnam. Our keynote speaker was Professor Fuqing Yang, Dean of the Faculty of Information and Engineering Sciences, Peking University, Director of the National Engineering Research Center for Software Engineering, and pioneering founder of the field of computer science in China.

The workshop was again organized around small working groups aimed at developing i18n curricular modules and laying the groundwork for international pilot projects and programs. One significant challenge is communicating internationally about scientific results in any field. One result of the efforts through this workshop was creation of a course on Scientific Writing in English for Computer Science Researchers. This experimental course was taught in Beijing Fall 2008 at Tsinghua University.

Workshop 3: Moving i18n Forward: from Ideas to Implementation

The third workshop was held in Portland, Oregon during May 2009. A keynote address on technical training and skill development programs for Google software and hardware engineers was presented by Maggie Johnson, Learning and Development Director, Google. Working group sessions included Cross-cultural HCI, Global Distributed Software Engineering, International Summer School, and Surveying Computer Science Education in the Pacific Rim. There were also short sessions on International Computer Ethics and a Model i18n Curriculum and Recommendations to IEEE/ACM.

As a result of these three workshops, a strong community of computer science educators was formed with several projects continuing into the future. There are two types of efforts underway: curricular innovations and international summer schools.

3. CURRICULAR INNOVATION

An initial action intended to incorporate the impact of internationalization into the computer science curriculum has been made in the interim ACM/IEEE review completed in July 2008 [3]. The review authors state: *With the increased globalization of computing and the international nature of the workforce it is increasingly important to pay attention in the curriculum to certain related matters.*

The international considerations mentioned in the review include international competitiveness, legal, social, etc issues, and cultural issues. Our experience points to a need for curricular innovation beyond this narrow set of issues, as discussed below.

3.1 Multicultural Communication

The issues in cross-cultural communication are not unique to computer professionals, but are certainly important in the context of the global economy, multinational development teams, and localization of computer applications. Awareness of other cultures and of one's own cultural viewpoint as a viewpoint rather than a universal reality belongs to the general set of skills students must develop. Other skills include language (with attention paid to cultural differences in use of "shared" English), and the process of cross-cultural accommodation. Difficulties of electronically-mediated communication are compounded by cultural differences.

As a common problem in cross-cultural communication we have identified the use of English as effectively the lingua franca of global IT. Native English speakers need to accommodate the difficulties of novice speakers and writers. Since translation can change the meaning of a statement, often frequent cross-checking is needed to ensure shared meaning. (For example, asking the other party to restate what they think was just agreed.) But there are also non-linguistic issues, such as Asian deference to the boss resulting in communication problems with superiors, or misunderstandings where "yes" should not be taken for an absolute answer.

There is a controversy about which culture determines the interaction: the professional IT culture or the national culture [1]? Some studies have shown remarkable consistency of values (motivation for work, views of authority, etc.) for IT professionals regardless of national culture. Other studies [2] have shown distinct national differences within the same professional culture (IBM).

We propose as a learning objective that students, after graduation, possess a set of skills that enable them to work effectively on multi-language, multi-cultural, distributed teams. These skills should include: Know how to work within good management practice and become a good manager oneself. (This mitigates cultural impacts.) Possess strong oral and written communication skills. Know how to assign responsibilities and meet one's responsibilities. While these objectives are not directly related to cross-cultural experience, they are a prerequisite to successfully tackling it. An important means towards acquisition of such skills is to participate in distributed, cross-cultural projects (see the following subsection). As a good starter project we propose work on distributed same-cultural projects (for instance, Free/Open-Source Software, FOSS). International experience, provided either in a more controlled environment by inter-university agreements, or by international internships are another avenue of skill acquisition. For those students with interest and time, departments should provide a track for spending time abroad.

It would be desirable to break down the learning objectives into, say, 6-hour modules, akin to structure of the IEEE/ACM Curricula 2001. These may be often achieved in the context of other computer science courses.

3.2 Global Distributed Software Development

Software today is often produced by global teams, but the conventional software engineering project course or capstone project provides little opportunity for students to learn the necessary skills for software development in a global setting. Students typically work in co-located teams where many of the problems of distributed development do not arise and peer-to-peer communication fills the gaps left by ambiguous specification, misunderstandings, and poor planning. Two overlapping working groups of the CPATH i18n project addressed issues of preparing students for leadership roles in projects that span continents, time zones, and cultures.

One of the working groups concerned itself with modules that could be incorporated into conventional courses in software engineering. The material to be covered in such course modules includes some straightforward practical skills, such as use of distributed version and configuration control systems and multiple forms of synchronous and asynchronous communication. More fundamentally, though, the challenges of globally distributed software engineering require a deeper grasp of core "computational thinking" [4] skills of appropriate separation of concerns, abstraction, and precise interface description. The structure of software artifacts and of the processes that produce software are deeply intertwined with organizational structure. As early as 1968, Conway observed that system structures inevitably reflect the communication structures of the organizations that produce them [5]; his observation was called "Conway's Law" by Brooks in his seminal book on software project management [6]. Parnas's pioneering work on information hiding in modular software structure [7, 8] is motivated by similar observations about the way human organizations conceive, construct, and evolve software systems.

A second working group, sharing several members with the first, considered development of courses in which students would actually collaborate internationally on software projects. A software project course typically tries to simulate, in a short project with a small team of students, the demands and constraints of a more critical project evolving over a longer period of time and with a larger and more diverse set of actors. The simulation is imperfect: The consequences of an inappropriate abstraction, with poorly placed module boundaries or poorly described interfaces, is far less when a homogenous group of students are co-located and have ample opportunity for discussion. A corollary of Conway's law is that modularity and abstraction matter much less for a small, homogenous, co-located development team. As a consequence, the success of a student project is often determined primarily by proficiency in coding alone, undermining the goals of a project-oriented software engineering course. The challenges of a real distributed project make it more difficult to substitute informal communication for abstraction and decomposition and clearly illustrate the interplay between separation of concerns in the software architectural design and organization of work, and between precise interfaces, appropriate formalism, and effective communication.

A handful of institutions around the world have already introduced global software engineering project courses, but there are many challenges to wide adoption, both technical and non-technical. For example, in addition to fundamental issues of working across institutional as well as national boundaries, something as simple as differences in academic calendars can complicate collaboration between student teams in different countries. The initial CPATH working groups recognized that nurturing development of internationally distributed project courses would require the joint efforts of a longer-lived international community, beyond the time frame of the initial CPATH project, but building upon the community established in that project. With support of a second CPATH project grant, a second series of community-building workshops has been planned to grow and maintain this community, along with two pilot offerings of internationally distributed project courses, one teaming students in Vietnam with students in the U.S., and another teaming students in China with students in the U.S.

3.3 Cross-Cultural HCI

We have developed a proposal for a 6 hour curriculum module to introduce undergraduate majors to the concepts of designing for cross-cultural human-computer interaction (HCI). The curriculum emphasizes the following major issues: i) Definition of cultural aspects of universal usability and developing software for international use; ii) Examples of the dominant interface culture and how it affects usability; Examples of culturally sensitive software artifacts; and iii) Cross-cultural software development with guidelines, human-centered development process and usability evaluation. The module will integrate into an HCI, software engineering, or general computer science course.

We see two areas of future activity. First is dissemination of our ideas and curriculum module. The proposed syllabus[3] needs be enhanced with a course description that includes

[3] Douglas, S. et al, Introducing Cross-Cultural Human-Computer Interaction (HCI) into the Curriculum, in preparation

readings, lectures, discussion material, and exercises. The goal of our dissemination plan is to make the CS community aware of the importance in incorporating cross-cultural HCI into the CS curriculum. This includes engendering broader discussion of the relevance of these results at conferences, and organizing workshops on teaching cross-cultural HCI to CS teachers. We have already started an online repository of resources and will continue to make the repository available for researchers and educators. Possible target partners include NSF Broadening Participation in Computing award recipients.

The second activity is the actual teaching and evaluation of the module itself or components of the module. Our repository of resources will allow new partners to both contribute new ideas and to evaluate our curriculum. We propose to make our model curriculum available to faculty members to try out at various institutions in order to solicit their feedback. We plan to seek out culturally diverse partners to help us to continue to understand the impact of culture on computers (and also the influence of computers on their cultures). We plan to utilize social networking tools to connect with our partners and to gather ideas.

Finally, we plan to make a recommendation to ACM/IEEE to include a curriculum module such as this one in the Computer Science curriculum under HCI in the next revisions of the Computing Curriculum, and to advocate for the inclusion of topics on internationalization in the curriculum and professional preparation.

3.4 International Computer Ethics and Social Impacts

The information age in which we live, with the use of networked computers located around the world, has created a global society. The subsequent mixing of cultures and economies has resulted in tensions regarding issues of personal privacy, intellectual property, freedom of speech, access to information, and computer security, among others. There are associated social impacts due to the changing nature of work, the development of online communities, and disruptions of established patterns of communication and control. The newly networked world does not automatically recognize national and cultural boundaries, leading to conflicts as to what is ethical and what are acceptable social impacts.

The internationalization of ethical issues and social impacts due to universal use of computers is the focus for the course module proposed here. Understanding the differing ethical and social expectations of people around the world represents an important element in realizing effective cooperation in cyberspace and will play a significant role in developing information systems that are acceptable and effective across cultures.

Many of these ethical issues have deep cultural and historical roots that go far beyond what can be covered in two weeks of study, as is anticipated for this course module. Therefore, the goal for this course module is to raise student awareness of the existence of cultural differences and the role they can play in the field of computer science, given its international context. To create such a course module, one must select issues to be addressed, review recent cases of controversy, determine the current state of affairs regarding selected issues, and collect materials that explain and discuss a variety viewpoints.

The outline we have devised for the course module is as follows: i) motivate discussion with recent issues in the news; ii) provide an international perspective by reviewing United Nations statements and ACM/IEEE codes; iii) discuss issue areas, by noting international agreements, national laws, cultural norms, business practices, and student attitudes and behavior. For the latter, we are developing questionnaires in various issue areas that students can complete at departments around to world to assess and illustrate differences in student attitudes and behaviors globally. The areas to be considered are intellectual property, privacy, information access, freedom of speech, and virtual communities. This module will expand upon what is usually addressed regarding these issues in a computer ethics course.

3.5 Computer Science Education in Asia

Enriching our knowledge of computer science education elsewhere in the world can inform development of U.S. curricula. Very little has been written about computer science education in the Pacific Rim. Together with our Asian partners, we are writing articles on CS Education in the Pacific Rim: China, Japan, Korea, the Philippines, and Vietnam.

As an example, we give some highlights from our SIGCSE 2010 paper on *Undergraduate Computer Science Education in China.*[4]

Established in 1956, China's flagship discipline, Computer Science and Technology, has become the largest undergraduate major in China with almost 800 departments nationwide enrolling over 430,000 undergraduate majors. While Chinese computer science education follows the ACM Computing Curricula 1991 guidelines, this model does not completely meet the economic needs nor the scale of the Chinese educational system. This need has been intensified as China moves from an elite education model to a mass education model. In order to address these challenges, the Chinese Ministry of Education has adopted a pyramid model of computing education that identifies three basic models for training the nation's high tech workforce: science-based education which prepares China's elite scientific talent pool for research-oriented careers; engineering-oriented education to train students in engineering design and implementation; and technology-oriented education focused on preparing a large workforce to produce applied systems to meet customers' needs. In addition, China is pursuing forums for dialog with international colleagues on advances in computer science education. In 2008, the first ACM Summit on Computing Education in China took place in Beijing; in October 2009, the Ministry of Education convened the 5th University Course Forum of Computer Science with keynote speakers Richard LeBlanc and Barbara Owens-Boucher, both leaders in the U.S. computer science education community. The next decade will be a time in China for innovation, international collaboration, and broader expansion of programs, as China evaluates the outcomes of recent educational initiatives and explores the future of computer science education in China and its role throughout the world.

4. INTERNATIONAL SUMMER SCHOOLS

In order to offer computer science students with immersive international experiences that fit into their heavy aca-

[4] by M. Zhang, Peking University and V. Lo, University of Oregon, *in press.*

demic schedules, we plan to establish an international summer school program: the *Pacific Rim Summer School in Global Software Development and Computer Science*. The summer school will provide a venue for faculty to implement new curricula in globally software development, multicultural communications, cross-cultural HCI, international computer ethics (as described in Section 3), and will provide students with the opportunity to work in cross-cultural teams and to take classes with students from around the world, in an intensive immersion-style experience. Our summer schools will bring students from both sides of the Pacific Rim to develop the hard and soft computer science skills that are needed to prepare them to work in an international context.

The anchor of our summer school will be courses with cross cultural content. These course modules are designed to be short (approx. 6 hours each) so that they fit into a one or two week summer school program. Students and faculty participating in inter-institutional global distributed software engineering projects during the academic year, will have a face-to-face meeting at the Summer School. In addition, participating faculty and industry representatives will offer courses and seminars in their own areas of expertise.

The first summer school is tentatively planned for July 2010 at Peking University's Wuxi Campus of the School of Software and Microelectronics. Located less than one hour from Shanghai, the campus covers an area of 500 Chinese acres and houses classrooms, research and development laboratories, dormitories, and faculty apartments. Dean Zhong Chen, Peking University, and Professor Andrzej Proskurowski, University of Oregon, head the Summer School Advisory Board which includes faculty and industry partners from the CPATHi18n Pacific Rim Community. For the first year, the Summer School will be limited primarily to CPATHi18n partner institutions. In subsequent years, we hope to make this opportunity available to a wider range of universities throughout the world.

In July 2009, we hosted the *International Summer School of Network Design and Operations*, on the campus of the University of Oregon, as a warm up to next summer's activities. This intensive hands-on course was attended by an international group of students from Africa, South Asia, China, and Oregon. The all-day classes were taught by staff from the NSRC (www.nsrc.org) and UO Information Services as part of the NSRC's mission to help develop and deploy networking technology to under-served parts of the world. After the week-long class, CPATHi18n students, who came from Tsinghua University and Peking University in China, took a four day trip with U. of Oregon students and faculty to visit some of our CPATHi18n partners in Oregon and Washington (Intel, Microsoft, Google, and the University of Washington).

5. SUMMARY AND FUTURE PLANS

We plan to synthesize the results of our efforts by developing a model *i18n* curriculum as a pathway through the computer science major. We will draft of a white paper recommending the addition of *i18n* topics to the IEEE/ACM Curriculum 2001 for the B.S. degree in Computer Science. This will amplify the recommendations of the IEEE/ACM Computer Science Curriculum Review Taskforce in July 2008.

We chose international collaboration in the Pacific Rim because of the growing importance of this region in the high tech industry and because our institutions have developed exceptionally strong Pacific Rim ties and expertise. We have worked within our regional context, but our outcomes should be globally relevant and long-lasting. The core of our developmental activities is a series of community building workshops involving a diverse group of academic and industrial partners across the Pacific Rim. The workshops addressed four key topics: assessing the needs, benefits and barriers to internationalization; cultural impacts on computer science courses and pedagogy; cultural awareness through international experiences; and community sustainability.

We are in the process of developing a globally available online resource repository for internationalization of computer science education[5]. This repository collects the results from our workshop series: seed ideas and recommendations for programmatic change, proposals for curricular innovation, new technologies for international pedagogy and collaboration, a database of international programs, key contacts, and bibliographic references.

We believe that we are well on the way towards effectuating the concept of internationalization of computer science education using the Pacific Rim as a model community. We have now a better understanding of the nature, challenges, and successful approaches for this transformation to be truly international in scope. Our projects contribute to a national discourse in this important arena.

The long term impacts of our Pacific Rim community-building project will be to give students at the participating institutions a competitive advantage in a global world, whether in research or industry. Internationalization of computer science education will help attract the best and brightest students and broaden the appeal of computer science to a much more diverse population. Computer science will be seen as a pathway to a career not in an isolated cubicle but in the wide-open world. Because scientific discovery is driven to a considerable degree by advances in computer science, students with global understanding of computer science will become better scientists, as well. Finally, internationalization will move our discipline towards the maturity and recognition it deserves, as more computer scientists move into leadership positions in commerce, education, and government.

6. REFERENCES

[1] Carmel, E., *Global software teams: collaborating across borders and time zones* Prentice Hall (1999), chptr. 5

[2] Geert Hofstede, http://www.geert-hofstede.com/

[3] Computer Science Curriculum 2008: An Interim Revision of CS 2001December 2008, ACM/IEEE.

[4] Wing, J. M., *Computational thinking*, Commun. ACM 49,3 (2006) 33-35.

[5] Conway, M. E., *How do committees invent?*, Datamation 14,4 (1968) 28-31.

[6] Brooks, F.P., *The Mythical Man-Month*, Addison-Wesley 2nd ed. (1995)

[7] Parnas, D.L., *On the Criteria to be Used in Decomposing Systems into Modules*, Communications of the ACM, 15,12, (1972) 1053-1058

[8] Parnas, D.L., *On the modular structure of complex systems*, in Proceedings of the 7th international Conference on Software Engineering, (1984) 408 -417

[5] http://www.cpathi18n.org