# A Broader View for Computing Education

*Rich Halstead-Nussloch, Mike Murphy, and Bob Harbort*
*Department of Computer Science*
*Southern Polytechnic State University*
*1100 South Marietta Parkway*
*Marietta, Georgia 30060*

**Abstract -** *In this paper, we describe a broader view for computing education. It has evolved in response to many influences, including opportunities for professional growth, requirements for retraining, curriculum guidelines, the needs of the State of Georgia and its University System, entrepreneurial characteristics of Southern Polytechnic, and the needs of business and industry. We are convinced this approach is a useful map for broadening computer science education and guiding individual citizens to chart a rational approach through an expanding maze of computing education options. Briefly, the spectrum described (and currently being implemented at SPSU) includes individual courses for narrow skill sets, continuing education certificates for well-defined domains, credit-based professional certificates for retraining, and the usual university degrees for traditional credentials. There are also longer threads of learning that a participant can follow within this computing education spectrum over time to increase expertise, credibility, and career mobility. Our guiding principles are based upon our experience with highly successful baccalaureate and masters programs:*

- *Graduates of our computing education programs must be citizen computer scientists who are competent, ethical, and committed to currency in their discipline through life-long learning.*
- *Computing education programs must provide clear academic and economic value to the citizenry.*
- *Computing education programs must provide meaningful credentials to business and industry.*

*A core factor in the success of our MSCS program is providing a transition path, as needed, in significant but manageable steps. Students with degrees from a surprising range of disciplines have successfully entered and completed our MSCS program. We provide opportunity, motivation, and guidance, which we now apply to a broader range of participants. In the worldwide information revolution with its drastic shortfalls in qualified computing personnel, one can either watch or play. We have chosen to play. How about you?*

## Introduction

During the 1990's we have all observed that computing competence has taken on a major importance in the industrial world. Computers have become interwoven in the fabric of virtually all human activity. For example, one of this paper's authors recently was called for jury duty. During the judge's selection interview, the response that his occupation was teaching computer science was met with a massed chorus asking, "Can you help us?" In this world, computing education takes on a major societal importance. In this paper we aim to share our experience and accomplishments in designing and developing a rational framework for computing education.

Because computing has become interwoven into all human activity, computing educators now have new stakeholders in what they do. Any framework that is used for design, implementation, or communication of education must be understandable by both traditional and new stakeholders. Traditional stakeholders include students, faculty, staff, administration, and professional and accreditation organizations. Many computer science departments also have brought industry and employers into the group of stakeholders. Of course, government has always been there in one form or another, but is now taking a more active and direct role. Citizens have taken on new roles: While the American Revolution was won with citizen soldiers, the information revolution is being won by knowledge workers building on the base of citizen computer scientists.

This year we developed a computing education framework that is useful for infusing rationality into the chaos of the information revolution. Since all computer science educators might benefit from our experience, we want to share our ideas in this paper.

### Background and History

In the past half-dozen years, computing education at Southern Polytechnic has achieved a number of significant milestones. Starting in 1991, we responded to a request from IBM to develop and implement a Master of Science in Computer Science (MSCS) program for working professionals that responded to IBM's needs. Currently, this program is available to everyone, and has grown to be strong in both quality and quantity. It graduates about 70 per year who go on to succeed in senior technical and management positions in metro Atlanta and beyond. Since many of our students were retraining into computer science, we found that transition courses were required to make this program work. This experience taught us that an important feature of quality computing education today is a transition path.

In 1996, we applied the principle of transition to design a new Master of Science in Software Engineering (MSSE) program. Although our MSSE degree requires a level of experience in computing for admission, we still make use of transition courses to bring students into alignment with the academic expectations at its base.

In 1997, we instituted a formal program to establish relationships with industry and provide certificates with professional credibility in programming and other skill areas. Through interviewing and surveying our industry partners, we identified a strong need for customized skill and professional training that would lead to such a certificate or degree. Again, this profile showed a strong need for mapping the transition to a recognized level of computing competence.

Our most recent opportunity to respond to evolving needs in computing education required that we extend these transition functions to map across all levels of computing education. To address these requirements, we developed the framework described below.

### Issues Addressed

Our experience has taught us that in order to develop effective curricula in computer science, we have had to address many other issues. Some of these issues are the traditional issues. Some are new issues or traditional issues that have manifested themselves in a new form. The primary issues we have chosen to address in this paper include:

*Society*: Because computing is central to everyday life and society, computing education programs must provide clear value to the citizenry.

*Commerce*: Because virtually all our graduates will need to earn a living, responsive computing education programs must provide clear value to business and industry.

*Professional and Personal Competence*: Because our graduates are the “product” of our educational program, they are the citizen computer scientists who are competent, ethical, and realize the need for currency in their discipline through life-long learning.

In summary, our purpose in this paper is to share a framework we have developed that adds rationality to computing education within the swirl of the information revolution. This paper also demonstrates the relevance and importance of responding to the evolving stew of societal, commercial, and professional needs.

## Influences and Requirements

Change in computing education occurs because of multiple factors. To understand our framework, it is useful to first understand influences that have driven our computing education program and curriculum development.

### Needs of an informed citizenry

To participate in the information revolution, citizens need appropriate knowledge and skills in computing. Since much of computing is complex, mastery requires time and study. The level and range of an individual’s computing mastery depends greatly on both the quality and the quantity of study. Furthermore, the criteria for mastery are constantly increasing, as the body of knowledge in computing is ever more rapidly expanding.

Citizens need to take a first step with computing, and then keep walking. To participate in the information revolution, people enlist by becoming involved in computing. To win battles in this revolution, they need the resources, skills and competencies to turn involvement into commitment. Citizen soldiers require training and education to be effective, and so do citizen computer scientists. Citizens thus need support from computing educators for both their initial experiences with computing and everything afterwards.

### Curricula and accreditation

The ACM and IEEE-CS provide resources and guidance on curriculum issues[1], and CSAB/CSAC provides guidance on program accreditation[2]. The regional accrediting agencies establish criteria to ensure quality in educational programs. The Southern Association of Colleges and Schools[3] accredits our university. Both types of organizations produce requirements for computing education. Since the ACM and IEEE-CS provide more specific direction, we concentrate on their 1991 curricula (while acknowledging the additional influence of regional and program accrediting agencies).

The 1991 ACM/IEEE-CS Curricula have the essential components of computing competence in a convenient structure. The guidelines start with nine subject areas, including Algorithms and Data Structures, Architecture Methods, Artificial Intelligence and Robotics, Database and Information Retrieval, Human-Computer Interaction, Numerical and Symbolic Computation, Operating Systems, Programming Languages, and Software Methodology and Engineering.

In addition to the subject areas, the ACM/IEEE-CS guidelines identify three core methodologies or processes for computing education:

- Theory, rooted in mathematics
- Abstraction, rooted in the experimental sciences
- Design, rooted in engineering

The subject areas and methodologies occur within a social and professional context. Computing-fluent citizens must also understand cultural, legal, ethical issues in computing.

Finally, the ACM guidelines name twelve recurring concepts that pervade computing: binding, complexity of large problems, conceptual and formal models, consistency and completeness, efficiency, evolution, levels of abstraction, ordering in space, ordering in time, reuse, security, and tradeoffs and consequences.

Using this structure as a basis, the ACM/IEEE-CS joint task force built its undergraduate curricula[1]. This structure was also used to develop the ACM Model High School Computer Science Curriculum[4]. The High School Curriculum is designed for transition to the undergraduate curricula.

Similar to the ACM/IEEE-CS, we have used this structure to define units of and pathways to computing competence. Computing education needs to adhere to the core areas, methods, and concepts for the discipline.

### Georgia and its University System

Southern Polytechnic is a part of the University System of Georgia. We must fulfill the mission mandated by our Board of Regents. We are funded by the Georgia State Legislature and also by the Georgia Governor's Initiatives. Our government provides both requirements and funding.

Economic development is a top agenda item in Georgia. A number of initiatives have been funded to increase the computing skills and competence in the state. Computing education needs to address these areas.

### Southern Polytechnic State University

Southern Polytechnic has a mission to fulfill, which includes supporting industry. It uses degree and other educational programs to do it. Some of these programs are non-degree certificate programs, others are continuing education programs. All programs help achieve a necessary continuum of opportunities for a quality education.

### Business and Industry

Today, any organization wishing to survive must establish and fulfill a valuable mission. Ours is to contribute to the economic vitality of Georgia. To do so, we must be responsive to the needs of business and industry. Our surveys have shown these needs include the following for computing education:

- Degrees and certificates with professional credibility
- Skill training
- Flexible logistics

### Requirements Summary

To be current and responsive in computing education, we found multiple requirements. They involve significant tradeoffs; for example, industry requires immediate skill training to meet specific needs, while the broad based ACM/IEEE-CS subject areas take time. At first glance, these requirements seem to be chaotic in profile, contradictory in substance, and too large in number to be all addressed effectively in a single computing education framework.

## The Framework

What appears to be chaos and contradiction at first glance sometimes turns into a blessing when put into the correct perspective. Indications are that a framework we have developed has such a benefit. Table 1 shows an attribute model of our framework with the four attributes spread across six levels of computing education. As one moves up and down the rows in this table, one can notice transitions in each of the four attributes.

***Table 1. Attribute-feature view of our framework for computing education.***

| Level of Achievement | Delivery Options | Focus | Degree Credit? |
|---|---|---|---|
| Doctorate | University | Professional | Yes |
| Masters | University | Professional | Yes |
| Baccalaureate | University | Professional (Entry ) | Yes |
| Professional Certificates | University | Competence | Yes |
| Skill Certificates | University<br>High School<br>Corporate | Skill Domain | No |
| Individual Courses | University<br>High School<br>Corporate | Single Skill | No |

Our experience indicates that the transitions between each layer in the table are key. For example, we have the most experience with the Baccalaureate to Masters transition. By designing this transition to be made in a number of significant but manageable steps, we have successfully brought people from all disciplines into computing.

Students with degrees in the arts, engineering, law, medicine, and sciences have all successfully entered and completed our MSCS program. Our transition courses cover many of the ACM/IEEE-CS essential components and recurring concepts that pervade throughout computing. The courses include:

- Advanced Programming and Data Structures
- Computer Architecture
- Operating Systems
- Mathematical Structures for Computer Science
- Object Oriented Programming in C++
- Database Systems

Our experience demonstrates the essential aspects of the ACM/IEEE-CS undergraduate curriculum form a good basis for an upward transition.

The high school curriculum also maps many transitions between the ACM/IEEE-CS undergraduate curricula and the high school level. Reports from a well informed high school teacher[5] indicate these transitions to be effective.

## Computing curriculum design building upon the base of the ACM/IEEE-CS Curricula

Our experience at the university level converges with the experience at the high school level:

- The 1991 ACM/IEEE-CS Curricula are an effective basis for designing content and structure.
- A well designed transition through the levels of Table 1 is critical to successful computing education.

This design principle also explains the Level of Achievement, Focus, and Degree Credit columns of Table 1. As we transition computing education downward from the Baccalaureate level, we can rationally award a professional certificate with degree credit. Degree credit is appropriate when the certificate is configured from existing, credit-bearing courses. Since a skill certificate can be delivered from outside the university, it makes sense that such certificates do not carry degree credit.

The Focus attribute refers to the scope of the educational unit's desired outcome. To be responsive to all constituencies, this results in a spectrum from a single skill all the way to a professional doctoral degree.

## A detailed view of our framework

Although it provides insight, the attribute view of our framework does not give the complete picture. Figure 1 shows our conceptual view of the major details. This motivates and illustrates key concepts for computing education design and implementation.

At its base, the chart shows its broadest band depicting the general citizenry. These people are not participating in the information revolution and therefore provide a recruiting pool for citizen computer scientists. It is in this group that our Georgia state government has expressed its keenest interest. In the post-information-revolution global economy, a country or state must have an adequate supply of knowledge workers—citizen computer scientists—in order to compete effectively. Thus, our framework is built on this broad base, and shows how citizens can expand their computing expertise through computing education.

The left side of the figure shows the prime sources of entry to each layer. On the right side, the figure shows the computing credibility level of each group. The first band of participants in the information revolution is entered by citizens or employees who take a formal computing course, generally of a continuing education nature. When citizens enter this band, they become citizen computer scientists.

This transition to a participating citizen computer scientist is best illustrated with an anecdote. One of the authors teaches in a continuing-education series of courses on C and C++ that is popular among Atlanta employers and general citizens as well. Students often come to the course after completing one or two of the "C/C++ in X days" self-study books. Although these students report they acquire skill through the book, they often state they were prompted to pay for the course because the self-study fell short. Many students state a desire to learn how to apply the language to multiple problem domains. Self-study appears to be effective within the problem domain of the book, but effective skill transfer to other problem domains requires interaction with an instructor. Thus our framework distinguishes between citizens and citizen computer scientists by that first, formal course in computing. To be effective this course should embody appropriate principles from the ACM/IEEE-CS Curricula.

Citizens will move up the ranks in the information revolution after taking that first formal course. One possible next step is to the traditional educational layers by applying to the university or college for a Baccalaureate or Associate Degree. This is a significant step for many people, especially if they already have a career or have made progress on their life journey after high school.

To better accommodate lifelong learners and industry, our framework defines two more educational layers, which straddle the university admission criteria. Beyond a course or two in computing, a citizen can become a paraprofessional by earning a Skill Certificate for a skill domain, e.g., networking or the C++ language. A step to the paraprofessional level does not require meeting university admission criteria, while the step above that to a

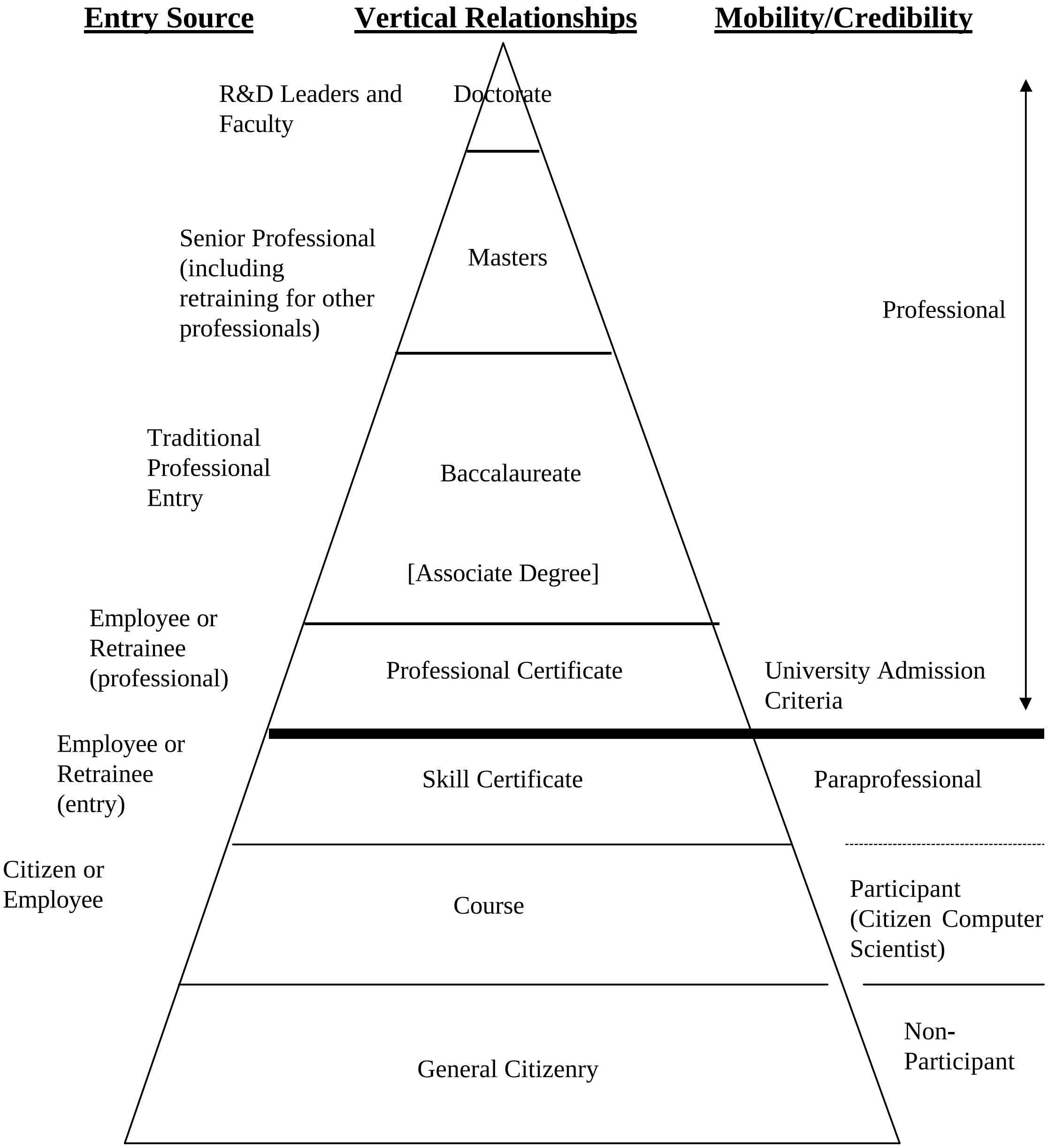

*Figure 1. Detailed view of our framework for vertically integrating computing education.*

Professional Certificate does. To earn a Professional Certificate, a student takes credit-bearing courses. Within our Georgia University System for example, we offer Professional Certificates in Programming, Information Technology, and Software Engineering.

Both levels of certificates are popular with industry, because they certify a level of skill or professional competence. The Professional Certificates have the added advantage of bearing university credit--an important professional credential. Employees and employers see the benefits.

In addition to industry finding these certificates an attractive innovation, so has our university system. Collaborating with other institutions, our university is developing and providing certificates attractive to employees and citizens alike. Funding the innovation is a special initiative of the University System of Georgia.

Our next conceptual step for the certificates is to investigate configuration of special certificates to meet the needs of individual businesses. For example, a company might need programmers who use Ada as opposed to C++, the current choice in the Programming Certificate. Ada could be substituted for C++ in the same curriculum.

By adding the certificates and building a layered approach, our framework establishes a rational map through the chaos of the information revolution. This map is easily understood by citizen computer scientists, computing professionals, industry, accreditation experts, faculty, and administrators. By following the ACM/IEEE-CS Guidelines where possible in all courses and certificates, every educational unit can contribute to a high level of computing competence in citizen computer scientists.

### Benefits of our framework

Our experience utilizing this framework has shown it leads to computing education that has an identifiable societal value to citizens. They can look at Figure 1 to see just where they are in the information revolution, and where they might go. It makes the commercial value of computing education obvious to employers and employees alike. The framework also provides a picture to show citizen computer scientists at all levels where they are in their lifelong career journey; it can reduce anxiety in making many of those transition decisions we all seem to be asked to make often in today's world.

## Summary Call to Action

With our innovative framework we are providing opportunity, motivation, and guidance, which should apply to a broad range of participants. In the worldwide information revolution with its drastic shortfalls in qualified computing personnel, one can either watch or play. We have chosen to play. How about you?

## References

1. "ACM/IEEE-CS Computing Curricula 1991," http://www.acm.org/educate/cc1991
2. "Criteria For Accrediting Programs In Computer Science In The United States, June 1996," http://www.csab.org/criteria96.html
3. "Criteria for Accreditation," Southern Association of Colleges and Schools, 1996
4. "ACM Model High School Computer Science Curriculum 1992," http://www.acm.org/educate/hscur
5. Tim Maley, personal communication