Connecting K-16 Curriculum & Policy: Making Computer Science Engaging, Accessible, and Hospitable for Underrepresented Students

Joanna Goode
Education Studies
University of Oregon
Eugene, OR 97403-5277
goodej@uoregon.edu

ABSTRACT
In this paper, a K-16 computer science reform effort is described as an effort that depends on curriculum development, professional development, and collaborative policy strategies.

Categories and Subject Descriptors
K.3.2 [**Computers and Education**]: Computer and Information Science Education – *computer science education, curriculum*.

General Terms
Human Factors

Keywords
Gender & Ethnicity, Wider Access, Pedagogy, CS Reform, CS Policy, Curriculum

1. INTRODUCTION
While there has been much research about the need to transform K-16 computer science education and broaden participation in computing, we have very few examples of large-scale K-12 computing programs that have successfully reached considerable numbers of girls and students of color. A university-district alliance is showing promising results for designing a joint curricular and policy approach for reforming computer science education.

This paper will highlight the parallel nature of curriculum development and policy strategy in providing an accessible and hospitable introduction to the dynamic nature of computing. The discussion will begin with a description of the development process of *Exploring Computer Science* - a curriculum designed to provide an innovative and engaging introduction to computing for students. The accompanying professional development emphasizes a student-centered instructional approach which reflects contemporary knowledge about how students learn. The results from a one-year pilot study suggest that this foundational course has a positive impact on motivation and interest in computing for diverse groups of high school students. However, without a policy framework to create the space for computing in the curriculum, opportunities for students to study computing will remain limited. Offering this course as a college-preparatory course in Los Angeles high schools requires endorsement from the state university system, school district, and high school principals. Navigating this complex system has only been possible due to the district-university collaboration which is driving this reform effort.

SIGCSE'10, March 10–13, 2010, Milwaukee, Wisconsin, USA.

2. THE COMPUTER SCIENCE EQUITY ALLIANCE
The Computer Science Equity Alliance first began reforming computer science education in Los Angeles Unified School District (LAUSD) in 2004. Charged with the results of a three-year qualitative research study which highlighted instructional and structural barriers to computer science education for girls and students of color [1-2], this alliance includes educational researchers, university computer scientists, K-12 school district officials, and leadership from the Computer Science Teachers Association (CSTA). Each summer from 2004-2007, the alliance offered a week-long professional development in the form of a Summer Institute for Advanced Placement Computer Science (APCS) teachers. An AP Readiness Program was launched so teachers can bring their students to the UCLA campus once a month for supplemental instruction and embedded professional development. These monthly meetings became a place for APCS teachers to continue to collaborate and build a teaching community. This approach of providing instructional and pedagogical resources and structural support to teachers led to staggering results – the number of students studying APCS in LAUSD tripled – girls quadrupled, Latinos quintupled, African Americans doubled [3].

However, formative and summative research identified a significant tension between the Alliance's goals of accessibility and the goals of the APCS course. Teacher interviews revealed that as a first course in computer science, the learning curve was too steep for the students to learn APCS content in time for the May exam. Teachers also reported that the course's focus on program methodologies in Java failed to attract the interest of students who were not already enamored with computing. It was clear from conversations with district officials and school administrators that the APCS enrollment increases were successful largely due to the college preparatory status of the course for college-bound students. This status is an important indicator of academic rigor, but also, students' course schedules are so impacted by graduation and college-preparatory requirements that they are often not able to take any non college-preparatory electives. It seemed that the success of the computing reform efforts were due to the academic status of the APCS course, not the particular content. University computer scientists joined this conversation and offered a host of computing topics that were more likely to tap into the interests of students. It became clear that in order to reform computing in Los Angeles, a new college preparatory course must be offered that introduces students to exciting introductory topics in computer science within an instructional framework informed by educational learning theories.

3. EXPLORING COMPUTER SCIENCE

Designing a curriculum for a new foundational content-area course requires deep knowledge in content, learning theory, and the communities from which students draw their informal knowledge about the world [4]. The resulting curricular materials reflect a selection of computing topics that are accessible and related to the academic interests of urban high school students.

3.1 Curricular Development Process

The development cycle for *Exploring Computer Science* sought to integrate these three types of knowledge into a curriculum designed for Los Angeles high school students. It was important to draw from many forms of local and national expertise in computer science education to inform the course content.

3.1.1 Instructional Framework

The instructional framework of the course was developed as a collaboration of university education researchers and LAUSD. Drawing from social constructivism [5] and sociocultural learning theories [6], this framework encourages a student-centered approach which draws from the informal knowledge students bring into each course. The course adopts an inquiry-based learning environment which requires students to engage in complex computing projects. Ongoing opportunities for collaboration, communication, and multiple ways of knowing are embedded into the design of the course.

3.1.2 Course Topics

The list of course topics for this curriculum was initially framed in an informal conversation at SIGCSE 2007 with leading K-12 and university computer science curriculum experts. With a general agreement of course topics and the sequence of these topics, educational researchers shared this list with LAUSD mathematics and science leaders and UCLA computer scientists who work closely with local computer science teachers and students. The scope and sequence outline was further revised after this additional input.

3.1.3 Developing Content

Five teachers, including two LAUSD teachers, contributed to the writing of curricular units in the course. Each of these teacher developers was provided the scope and sequence of the entire course, the instructional framework, and a list of objectives for their particular unit. The teacher writers drew from their own knowledge and teaching experiences to contribute towards a particular computing topic. The authors of the curriculum integrated these ideas into a cohesive, six-unit curriculum designed to take place over a two-semester school year. Daily teacher instructional plans and supplementary resources are included in this curriculum.

3.1.4 Training for Pilot Teachers

Six teachers came together for a pilot professional development session in 2008. These teachers were later joined by 25 prospective teachers twice during the school year for continued professional development. The professional development opportunities were carefully designed to highlight the instructional strategies and the content of *Exploring Computer Science while* providing a format for feedback about the curricular materials.

3.1.5 Piloting of course

A total of seven teachers piloted the course during 2008-2009. Almost 300 students enrolled in the pilot courses across these seven comprehensive high schools. Females accounted for 42% of course enrollment and African American and Latino students made up 92% of course enrollment. Semi-structured teacher interviews, teacher surveys, student surveys, and observations informed specific and general revisions that were needed in the curriculum and a major overhaul took place in 2009. The pilot study revealed promising findings about the effectiveness of *Exploring Computer Science* for students. Using pre- and post-survey instruments, research showed that participation in the course resulted in:

- changed students' perceptions of the usefulness of computer science in a positive direction ($p<.05$);
- changed students' beliefs about the appeal of computer science in a positive direction ($p<.01$);
- increased students' perceptions of computer science as enjoyable and stimulating ($p<01$);
- increased students' motivation to stick with a difficult problem ($p<.001$);
- students' willingness to have the answer given to them instead of working it out decreased ($p<.05$);

Further, students reported that they were more likely to take computer science courses after high school and pursue computer science as an academic major after enrolling in the course. These results point to the impact of *Exploring Computer Science* in shaping urban students' understanding and appreciation of this discipline.

3.1.6 Distribution of Materials for Course

The release of *Exploring Computer Science* took place prior to the summer professional development in 2009. An early release of this curriculum allowed an exploration of the materials before the 2009-10 school year embarked. The curricular materials have been released to the computer science education community through the CSTA website [7].

3.1.7 *School District Expansion of Course*

The final step of this curricular process was the buy-in of the school district to offer this course. Armed with the university support to offer professional development and ongoing instructional support in addition to free curricular materials, the Los Angeles Unified School District utilized its policy influence and professional networks to bring the course content to classrooms across Los Angeles.

3.2 Content of *Exploring Computer Science*

The resulting course is composed of six instructional units: 1) Human Computer Interaction; 2) Problem Solving; 3) Web Design; 4) Introduction to Programming; 5) Robotics; 6) Computing Applications. The course reflects topics and objectives drawn from Levels II and III of the ACM's K-12 curricular framework [8]. Each unit begins with a discussion of the final project so students have an appreciation of the problem they are working towards solving while learning the necessary skills required to solve the problem. Assignments and instructional strategies weave through the six units, conceptually linking the topics together. The curricular materials include original assignments as well as lessons drawn from other computing education materials, including Scratch [9] assignments, Lego Mindstorms robotics challenges, and Computer Science Unplugged [10] activities.

3.3 Professional Development for Course

The professional development designed to accompany the curriculum is an essential companion to the *Exploring Computer Science* course. In order to offer the course at a school site, principals must send a teacher to the ECS Summer Institute and weekend workshops. Participating teachers receive a stipend for their participation in the professional development events.

In 2008, six teachers attended the ECS Summer Institute in preparation for teaching the pilot course in 2008-09. In 2009, 23 teachers from 20 schools participated in the Summer Institute. These culturally diverse teachers represent the geographical diversity of the Los Angeles basin. As the first year of district-wide *Exploring Computer Science* formally launches in 20 Los Angeles public high schools in 2009-10, teacher reflections of the preceding ECS Summer Institute radiate a sense of passion and justice for providing accessible computer science education to students of color attending large, public schools. Teachers report a shared sense of community amongst one another and an awareness of explicit pedagogical approaches that create places of academic hospitality.

The goals of the professional development programs include but reach far beyond the learning the computing content included in the instructional materials and extend building instructional strategies that support learning in a computing community of practice. Such goals, however, require ongoing professional development opportunities.

3.3.1 *Modeling Engaging Pedagogy*

Given the instructional design of the curriculum, it is important to explicitly model the inquiry-based design of the course in teacher workshops. To this end, several instructors take turns facilitating the professional development activities and publicly reflecting on their own teaching. As a result, teacher participants report that their deepened understanding of how computer science content can be taught through role playing, jig sawing activities, pair and small-group collaboration, structured tinkering, multiple solutions, utilizing manipulatives, simulations, English language learner modifications, journal reflections, and interdisciplinary connections. Rather than providing the answers, the instructors facilitated participant discussions of problems and drew from the rich and varied expertise in the classroom.

3.3.2 *Computing Community of Practice*

A unique feature of the professional development is the distributed nature of expertise about teaching, computing knowledge, and the culture of Los Angeles high school students. No one participant or instructor in the professional developments has the singular solution for broadening participation in computing; rather, the professional development becomes a space where conversations about teaching wisdom and innovative computing topics intersect. Teachers, some who have been teaching APCS for years, join other teachers who have powerful pedagogical ideas but need to build their computing content knowledge. University computer science educators and veteran practitioners facilitate the conversation and continue to learn from the classroom teachers who connect with students daily. Important professional relationships have been built amongst co-learners and a computing community of practice has been cultivated and continues to grow with each subsequent workshop. Evaluations of teacher participants point to the desire to bring this academic accessibility and hospitality to their own classroom community.

3.3.3 *Ongoing Professional Development*

Since the ECS Summer Institute is only a week-long, three follow up Saturday sessions are held during the school year to focus on the last two and a half units of the curriculum. This ongoing model of professional development provides teachers a chance to come together and reflect on their classroom teaching experiences while learning new curricular material.

Newly hired computer science coaches, provided through additional NSF support, have begun following and supporting these teachers into their classrooms to provide content tutorials, co-teaching opportunities, collaborative planning sessions, and instructional support as needed.

4. POLICY STRATEGY

Though the development of the *Exploring Computer Science* course underwent a rigorous design process, the accompanying policy strategy was essential for ensuring the integration of this course in the school curriculum. New courses require the support and buy-in of school and district administrators. Though each school system is unique, the process of building an accessible program of computer science education in Los Angeles high schools provides lessons and considerations for other school systems interested in reforming computer science education. Due to the inter-connected and bureaucratic nature of state, district, and school policies, these efforts require a strategic multi-tiered approach to integrate this curriculum into the classroom setting so it is accessible to all students.

4.1 State University Level Policy

In 2008, California released a call for the development of new college-preparatory courses which merge the topics of traditionally academic courses with the applied nature of Career Technical Education (CTE). University education researchers collaborated with district officials to submit an application to the University of California Office of the President to deem *Exploring Computer Science* a college preparatory elective course. The first year there was a question about prerequisites for the course and the district was encouraged to resubmit the application with a preparatory course in place – a full year later. The resubmitted application reflected Algebra I as the pre-requisite for this course. The University of California Office of the President has approved this college-preparatory application with glowing reviews, noting that this course is "a very strong course submission" that they would like "to use it as a model CTE course". This means that the course satisfies multiple purposes for students on various pathways towards college, careers, or other directions.

4.2 District Level Procedures

The administrative structure of the school district relies on official, school board-approved memorandums to superintendents and principals in order to formally provide any notices in the district. To publicize this course and gain principal approval, school district Equity Alliance partners work internally to navigate this process and develop informative memorandums about the course and accompanying professional development opportunities. Without this insider status, maneuvering within this complex system would be difficult given the tight resources and multiple demands of school districts.

4.3 School Level Implementation

These official district memorandums were followed up with individual meetings and conference calls with high school principals. Principals who express a signed commitment to offering *Exploring Computer Science* were asked to identify a dynamic teacher able to attend the extensive accompanying professional development. Importantly, one of the Computer Science Equity Alliance's leaders is the former LAUSD Director of Secondary Science and current school principal of one of the largest LAUSD high schools (which serves the largest number of African Americans in the district), and another Alliance member is now the secondary science director for LAUSD; together they are able to draw on an extensive professional network to offer ECS to principals and place *Exploring Computer Science* in a growing number of schools.

Once connected with the CSEA community, teachers are provided support and resources (counselor information forms, student recruitment forms, parent information, posters, etc.) to further advocate for computing courses at their school site. Teachers are in an important position for working with counseling offices and other educators to advocate for the steering of traditionally underrepresented students towards *Exploring Computer Science.*

5. A CASE OF LOCAL REFORM

Though the particular details of reforming computer science education outlined in this paper are unique to Los Angeles schools, a select set of the procedures and strategic choices made to broaden participation in computing in this diverse city might serve as a model for other collaborative reform efforts.

5.1 Policy Work is Collaborative

Securing bureaucratic approval for a new course in the second largest school district, and in the largest state, requires a careful navigation of school system demands and opportunities. Each policy move was a result of collaborative decision making of district officials, educational researchers, university computer scientists, professional organizations, and lead teachers. The Computer Science Teachers Association advises this project and situates it within a national context of computer science education. University computer science educators were instrumental in providing content guidance while educational researchers brought in a framework for curricular reform. In addition to the value of multiple perspectives on policy decisions, several of the endorsement applications require district-university partnerships. The strength and successes of the Equity Alliance are a testimony of the power of collaboration. However, there are limitations to shaping only district and state stances on computing. National policy which incorporates computing into education legislation is a necessary step in encouraging state and district leaders to support computing education and intensify the impact and sustainability of local reform efforts [11].

5.2 Example of Dynamic Foundation Course

As the College Board reconsiders the APCS programs, there has been much discussion about a sequence of courses in computing. The decision to offer a survey course of major computing topics as a high school computing course was based on a desire to introduce students to the broad field of computing through an exploration of engaging and accessible topics. The curricular materials offer structured activities and projects which allow students to dabble in the computational thinking practices championed by computer scientists [12]. It is anticipated that some students will use this course as a launching pad to a soon-to-be revised APCS program.

Though curricular development is most effective when created as part of a local effort, many districts and states do not have the capacity to design their own curricular materials. Though developed with the particular needs of Los Angeles students in mind, these curricular materials could certainly be adapted nationwide to supplement other instructional materials. Apart from the cost of robots, this course does not rely on expensive computers, a particular platform, proprietary software, or costly textbooks – making it an attractive option for schools, districts, and states struggling to provide accessibility with scarce resources.

The organization of the *Exploring Computer Science* curricular materials makes each unit amenable to dynamic substitutions or modifications as engaging new computing topics are brought into the curriculum. For example, a Computing Applications unit currently draws from seismic data provided by UCLA's Center for Embedded Network Sensing (CENS), a member of the Computer Science Equity Alliance. As the CENS data evolves, or large-scale application projects are developed elsewhere, these changes will be added or interchanged in the curriculum. This structure allows more local connections to be included or emphasized based on the needs of students. Importantly, this dynamic design is necessary for students and teachers to keep up with the rapid pace of computing innovations.

5.3 Teachers are Key

The close link of the curriculum to professional development is a necessary coupling for introducing a new course in high schools. In this model of reform, the effectiveness of the course is limited if not accompanied by high quality professional development which focuses on content and instructional strategies. Though research has shown that the ECS Summer Institutes and weekend workshops are helpful in supporting *Exploring Computer Science* teachers, the data also shows a need for more personalized mentoring. The new computing coaching program will seek to answer that need.

Working with teachers is critical because the implementation of reforming computer science education ultimately takes place in the classroom setting. Observations from this project have captured powerful teachable moments when skilled computing teachers link students' knowledge and interests with computing topics – such as JavaScript and MySpace page design. Many of the Los Angeles teachers in computer science share a common vision of inspiring youth. In addition to focusing their passion, beauty, joy, and awe on the computing *content*, these teachers focus their passion, beauty, joy, and awe towards the enormous *potential* of their computing students. This subtle shift in perspective – from *teaching content* to *teaching students through content* – emphasizes a critical distinction in purpose.

5.4 Measuring Success

Sharing a goal of broadening participation in computing, the Computer Science Equity Alliance will continue to work to improve the engagement, accessibility, and hospitality in computer science education. Collecting quantitative and qualitative data to measure students' engagement with computer science is integral to evaluate the effectiveness of this model of reform. However, the Equity Alliance resists the call to make the longitudinal impact of this course on college major patterns as the primary evaluation criteria. The measurement for success differs: For students to have access and understanding of computing as a result of this course is considered a successful effort by the Equity Alliance regardless of students' future academic ventures. It is opportunities, not just a narrow end goal that the Equity Alliance seeks to provide.

6. ACKNOWLEDGMENTS

This work would not be possible without all the contributions from the members of the Computer Science Equity Alliance. This work is supported by the National Science Foundation. I would also like to express appreciation to Ruwan Jayasinge, a passionate Los Angeles computing teacher, who coined the term "academic hospitality" to describe a Summer Institute experience.

7. REFERENCES

[1] Margolis, J., Estrella, R., Goode, J., Holme, J. J. and Nao, K. *Stuck in the shallow end: Education, race, and computing*. MIT Press, Cambridge, MA, 2008.

[2] Goode, J., Estrella, R. and Margolis, J. *Lost in translation: Gender and high school computer science*. MIT Press, Cambridge, MA, 2006.

[3] Goode, J. If you build teachers, will students come? The role of teachers in broadening computer science learning for urban youth. *Journal of Educational Computing Research*, 36, 1 (2007), 65-88.

[4] Bransford, J., Brown, A. L. and Cocking, R. R. *How people learn: Brain, mind, experience, and school*. National Academy Press, Washington, D.C., 1999.

[5] Vygotsky, L. *Mind in society: Development of higher psychological processes*. Harvard University Press, Cambridge, MA, 1978.

[6] Wenger, E. *Communities of practice: Learning, meaning, and identity*. Cambridge University Press, Cambridge, UK, 1998.

[7] Goode, J. and Chapman, G. *Exploring computer science*. Computer Science Equity Alliance, Los Angeles, 2009.

[8] Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C. and Verno, A. *A model curriculum for K-12 computer science: Final report of the ACM K-12 task force curriculum*. Association for Computing Machinery, Computer Science Teachers Association, New York, NY, 2003.

[9] Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M. and Rusk, N. Programming by choice: Urban youth learning programming with scratch. In *Proceedings of the Proceedings of the 39th SIGCSE technical symposium on Computer science education* (Portland, OR, USA, 2008).

[10] Bell, T., Fellows, M. and Witten, I. *Computer science unplugged!* , City, 2009.

[11] Wilson, C. and Harsha, P. The long road to computer science education reform. *Communications of the ACM*, 52, 9 (Sep 2009), 33-35.

[12] Wing, J. M. Computational Thinking. *Communications of the ACM*, 49, 3 (March 2006), 33-35.