

# One size fits all?! One size fits none!

## A Custom Computer Science Education Proposal

Daniel D. Garcia  
EECS Department, UC Berkeley  
ddgarcia@cs.berkeley.edu

*“You can have any color you want, as long as it’s black.” – Henry Ford*

As educators, we have three important, persistent challenges at the introductory level: diversity, diversity, and diversity.

We need diversity in our student body; the percentage of women and students of color who finish our programs is heartbreaking. They have the highest attrition rates of any other group, and there are few signs of change. As our enrollments continue to wane since the boom of the past decade, the relative number of these students has dropped even further; it seems they were the first to go. How do we *encourage* them to sign up, how do we *nurture* them once they are here, and how do we successfully *launch* them into the real world?

We all struggle with diversity of student background in our gateway courses. How do we assess the prior preparation and current state of a student’s understanding in order to adjust our program to match the *impedance*? In an ideal world, their first lecture would start with the gentle introduction: “Welcome! In your last high school / college course, Mr. Hernandez was discussing X, now let’s see where we can go next!”. Even *given* a detailed work-up for a student, do we have the *ability* to tailor our courses to maximize Piaget assimilation?

Once our students are successfully rolling in our course, how do we address the diversity of interest, ability and learning styles? We have some students who are taking our course because someone told them there was “gold in them thar’ majors”, some seeing non- { web, IM, email, iTunes, illegal BitTorrent } computing for the first time, some who might as well have a scarlet B on their chests, and some who will someday sit beside us in the Ivory Tower. How do we hold the interest of the first group, provide a slow-but-steady learning curve for the second, push the third and hold the interest of the fourth while inspiring everyone *and* fulfilling our syllabus post-requisites?

*“You want to go where everybody knows your name, and they’re always glad you came . . .” – theme song from Cheers*

We live in an era of portability and limitless customization; if the sixties was the Age of Aquarius, this is the Age of iPod. Our students’ cell phones, mp3 players, and Palm devices know their preferences, favorite songs and recent phone calls. They shut down in an instant and pick up right where they left off. They enable flavor-of-the-day customization. They ask the user: “where do you want to go today?”, and respond semi-intelligently. They are lightweight, cheap, have great designs and are a lot of fun.

The typical uninspired CS intro course has *none* of this and issues a single, drab grey, Acme™ one-size-fits-all jumpsuit to each student. Different *interests*? “They’ll learn about possible applications in the upper division.” Different *abilities*? “If you’re having trouble, re-read the chapter. If you’re bored, do both the odd *and* even exercises in the back of the book.” Different *learning styles*? “You’re going to learn the way we teach and you’re going to like it! Now, eat your broccoli.” There’s also enormous overlap and duplicated effort with our instruction; every high school in the country is reinventing the wheel *every* semester teaching students AP Java, for example. There should be a vast “resource well” (ala *Nifty Assignments*<sup>1</sup> but with more dimensions) where we could blend the best practices into a big soup from which everyone interested could drink. Reduce, reuse, and recycle!

There are tunnels of hope in this mountain of despair. The education community has rallied behind the “guide on the side” vs. “sage on the stage” model. Here at UC Berkeley we’ve built an online, lab-centric, collaborative system called *UC-WISE*<sup>2</sup> that allows students to go at their own pace, although they need to sync with the rest of the class each week. The system remembers what they’ve completed, where they’ve left off and jumps right there when they log in. It allows for “town hall” collaboration, e.g., discussion groups and brainstorming. One of the keys to its success is the tight coupling of a new topic to (say) a short exercise, then to a discussion, then a quiz and finally to a homework problem, all the while supported by a team of rabid lab assistants working the room ready to help with any question. *TuringsCraft*<sup>3</sup> is another attempt that gets close but misses the mark. I summarize my vision of the *ideal* system below:

---

<sup>1</sup> <http://nifty.stanford.edu/>

<sup>2</sup> <http://www.ucwise.org/>

<sup>3</sup> <http://www.turingscraft.com/>

- **Online.** In an era where light bulbs will soon have their own IP addresses, this is a no-brainer.
- **Free.** Any student in the world with a computer and an Internet connection should be able to have *full access*.
- **Lightweight.** One shouldn't need a fast box or a fat pipe for basic services. The system should feel nimble and responsive.
- **Fun.** The system should have a sense of whimsy. Computer science is not about coding alone in a basement lab anymore; we have to hold the interest of the MTV generation who has very high expectations. If it's delightful, retention becomes a snap.
- **Open-source to the core.** Both the *message* (content) and the *messenger* (system) need to be open to the community, though instructors and students will probably have different levels of access. Super-users will monitor abuse.
- **Channels.** Some want Java objects first, some want objects later, and some have seen the light and teach *Scheme*. ☺ There needs to be enough flexibility to easily support, edit and configure these self-clustering communities.
- **Meta-question authoring.** One needs the ability to author, say, a sorting quiz question at an abstract level, and when the student takes the quiz, the particular keys would be generated on the fly. This would allow plagiarism-free online exams.
- **Collaborative.** Students should know they aren't alone. There should be tools to support partner-based work, group discussions, brainstorming, blogs, online study sessions, shared whiteboards, wikis, remote instruction, guide-on-the-side help, etc.
- **Multi-modal.** There are as many different learning styles as there are fingerprints. Some like to hear it first in lecture, then do a small problem before reading the book. Some eschew lectures and jump right into the projects. The system should have many parallel ways to teach the same thing and give the student the choice. This would include static pages, adaptive online questions, snippets of lecture videos from local and/or remote lecturers, multimedia, algorithm animations, etc.
- **Adaptive.** The new SATs (I'm told) and graduate school oral qualifying exams have been doing this for years. Students start at the pace of their choice based on their self-measured ability or interest. If they start to ace questions, the difficulty ramps up. If they make an error, the questions get easier with the overall goal to figure out the limit of the student's knowledge.
- **"Big Sister".** This is an attempt to spin "Big Brother" as a *good* thing. For example, you (the student) walk into your first CS course and volunteer to take an extensive adaptive assessment, geared to understand who you are and what you understand (note I didn't say "what you were taught") which is stored in your profile. As you proceed through the course, it learns what is difficult and what is easy for you. It learns how you like your content delivered. It learns your schedule, knows the deadlines and can determine the pace appropriately. If used in several courses, it knows what you've absorbed (including what you should have but haven't!) and provides refreshers. Instructors can perform queries to determine what to review.

*"We make change. If you give us a quarter, we'll give you five nickels. We'll give you 25 pennies. We'll work with you. People ask us how we make money. One word: volume." – Saturday Night Live commercial spoof*

The system would be available as a free resource to every intro course in the country. Each student would have their profile stored and referenced by *Big Sister* as they moved through the curriculum. At any stage a history of achievement or confusion could be made available for targeted human tutoring. E.g., an instructor could query student profiles and pair a recursion ace with one who is struggling.

Greed, copyright, inertia and intellectual property are the biggest inhibitors. Instructors are very conservative with changes to their curricula, and rightfully so – great care has gone into them. SIGCSE-coordinated, open source (and NSF-funded) initiatives are the keys to success. Separately, we're all reinventing (or teaching) the wheel every semester; it's a cacophony. *Together*, we can pour our ideas into a vast resource as has been done with the *Nifty* work. Meta-data would be the key to organizing all of the disparate information; we'll need to be able to perform meaningful searches. Institutions of similar size, curriculum or pedagogic leanings could cluster to form a channel.

Whoever joins the overall design team needs to have a vested interest in the outcome. Moderators would manage the content within each channel and project administrators would oversee improvements to the delivery interface. Government could fund the initial prototype and architecture, perhaps leveraging off of *UC-WISE*, *TuringsCraft*, the *Nifty* archives, and the wonderful *Cut-the-knot*<sup>4</sup> site (or all four!). Professional societies could provide the umbrella forums for the evangelism, interface usability & self-evaluation discussions and help to pave the road to adoption. University faculty would author content and, as the Borg say, "add their uniqueness to the collective".

This paper outlines a long-term, high-impact strategy to address what is arguably the biggest problem with introductory computer courses: *diversity*. It proposes a customizable learning environment, in which students are allowed to follow the path and learning style that best suits them *within* a course – it works with whatever fixed curriculum has been chosen. The novel idea is that the system tries to track the student's progress and adaptively change, pushing those who are idle and pulling those who are falling behind. *Big Sister* is much more than just an annoying animated paper clip – it is a personal tutor that can empower students by adapting to *their* needs.

<sup>4</sup> <http://www.cut-the-knot.com/>