

# Graduate Education: An Industry Perspective

**Jonathan Grudin**  
Microsoft Research  
Redmond, WA 98027  
jgrudin@microsoft.com

## ABSTRACT

After close to a decade of teaching graduate HCI classes, I moved to an industry research lab that hires PhDs in HCI and other areas. I have close ties to product groups that hire people with MS and PhD degrees in HCI. The ecology of industry is getting more complex, presenting challenges for academic programs. Meanwhile, academic research has its own trajectories. These may not match closely, as suggested by decreasing practitioner participation in CHI, despite the growing number of practitioners with advanced degrees.

## Author Keywords

Curriculum, research, practice, design, development

## ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

## ONE INDUSTRY PERSPECTIVE

I taught graduate HCI and CSCW courses in primarily computer science departments from 1989 through 1998 at five universities on three continents; primarily at UC Irvine but also at Aarhus, Keio, Oslo, and the University of Washington. In 1998 I joined Microsoft Research, where I have collaborated with researchers and people with advanced HCI degrees in the product groups; I published several papers with people in product groups. My wife Gayna Williams manages a large “user experience” group, comprising twenty people with advanced degrees in HCI-related disciplines. Both research and product groups hire student interns, which are further opportunities to examine the match of academic training to industry orientation. Of course, Microsoft is not typical.

In the past decade, industry has changed dramatically. Emphasis has shifted from narrowly defined usability to ‘user experience,’ for which usability testing is only one component. Several factors contribute to this shift. As computing

is more thoroughly integrated into people’s lives, use must be considered as part of a larger interconnected whole. More applications and features support communication, information sharing, and coordination. Visual design, interaction design, branding, marketing are all part of much user experience and have to be considered together.

The pace of technology change is quickening. Moore’s Law now delivers huge increases every 18 months or so. Entire industries are eroded in succession: film cameras, diskettes, audiotape, videotape, and so on. In addition, growing numbers of students, companies, developers, and entrepreneurs around the world translate into rapid advances for any hot topic. Not long ago most innovations were incubated in universities and spun out into startups; it isn’t clear that that model is enduring.

The level of HCI expertise in our company is far higher than it was a decade ago. Although Microsoft may be atypical in this regard, graduates are in general entering a world where many people have five, ten, or fifteen years of day-in, day-out experience doing user studies or user research.

HCI programs have mostly arisen in Computer Science departments. In the beginning, more were in or associated with psychology, which provided more grounding in methods of behavioral research. Computer Science generally emphasizes building a system. HCI contributes by encouraging requirements analysis and insisting on empirical study of use, but to have a chance of conducting a realistic test, a student must do formidable system-building. This comes at the expense of developing assessment skills, and the studies generally prove relatively rushed and inconclusive given the challenges of finding a site and encouraging use in a constrained time.

In our environment, division of labor translates into a low demand for developer skills in those doing user studies and analysis or interface design. Students who take a small number of HCI courses may position themselves as program managers, who negotiate specifications among high-level managers, developer, designers, and the user research teams. But they don’t have the strong grounding in methods expected of those hired to do user and usability research.

Many usability methods and other user research methods employed in industry have now been used heavily and experimented with for many years, so there is far more specialization. Dense books on paper prototyping, scenario construction, persona use, contextual design, and other ap-

proaches are available. Employment of trained ethnographers is more widespread. The ability to collect phenomenal quantities of data using the Internet has made data mining a hugely promising tool. In our context, user experience professionals must work alongside designers and marketing people as well as developers. Where are these methods and skills to be taught?

### **The HCI PhD**

Graduate students in many Computer Science programs have requirements that in some ways work against what is needed of usability or user experience researchers or interaction designers at Microsoft. They must cover a range of computer science course work that makes it difficult to get a solid grounding in any one behavioral method, much less a sense of expertise with a range of methods. They usually must build (or want to build) a system, another large investment that has some value in interacting with developers but is not highly valued in environments where specialization is pronounced.

What they are generally exposed to in course work are the ‘discount usability’ or ‘quick and dirty’ or ‘heuristic’ methods that are in fact heavily used in industry, but they are usually not exposed to formal experimental studies, detailed statistical analysis, and so on. Although formal experiments and statistics are not widely used in industry, people with those skills are more likely to understand what they are trading off with the discount methods, what it is that they can and cannot claim based on limited studies. They will be better prepared to respond to challenges to their results from the occasional team member who may understand statistics a little better than they do. Certainly, to prepare for this environment an HCI program should include strong training in methods. It is possible that Information Schools will provide an opportunity to shift the balance of training to include more behavioral methods and analysis. People with a small number of HCI courses are often valued at MS, but in the ‘program manager’ position, responsible for specifications and for coordination between user research or usability, design, development, and testing.

Graduate students who pick an interesting, important area and spend three years building and deploying and testing a system also face a growing challenge from the rapid increase in technology advance over the past five years. There is a fair chance that through the efforts of some of the hundreds of thousands of eager professional and amateur developers out there, often working in teams with better resources and more experience, the world will have overtaken their system. Some teams may be tackling their problem before they begin. I recently had what seemed an original insight, not discussed in the research literature. I mentioned it to a young colleague in a product group who directed me to something very similar described in someone’s blog in 2001, before I was particularly aware that blogging existed.

I am on the PhD committees of good students in several university CS departments with strong HCI programs. All have taken the approach of collecting requirements-flavored data, analyzing, designing and building a system, deploying it, studying its use, coming away with ideas for redesign or further research. I have seen the challenges outlined above. When I described it to my wife Gayna Williams, who in the course of managing a large usability group has interviewed dozens of people graduating with PhDs and MSs in HCI-related disciplines, she told me how she addressed this common scenario. She looked for graduates who could describe well how the field had changed as they worked, and how they shifted in response to changes in the environment. John Gould emphasized at various times a decade or more ago that a key skill required in our field was to be willing to work in a ‘sea of change,’ and that seems even more true today. It is tempting to think that at a time of rapid change, understanding history is not as important, but it is a step toward thinking in terms of dynamic change.

### **REFERENCES**

1. Grudin, J. (2004). Crossing the divide. *ACM Trans. on Computer-Human Interaction*, 11, 1, 1-25.
2. Grudin, J. (in press). Three faces of human-computer interaction. *IEEE Annals of the History of Computing*.