Kafai, Y. B. & Roberts, M. (in press). On Becoming Junior Software Designers. In R. Steevens & P. Bell (Eds.), *Proceedings of the Fifth International Conference on the Learning Sciences* (pp. 191 – 198). Seattle, WA.

On Becoming Junior Software Designers: Students' Reflections on their Participation and Development in Collaborative Software Design Activities

Yasmin B. Kafai and Meridith Roberts
University of California, Los Angeles, 2331 Moore HallLos Angeles, CA 90095-1521
Tel: 310-206-8150, Fax: 310-206-6293
Email: kafai@gseis.ucla.edu

Abstract: This paper examines the identity development of 12 students who participated in several collaborative software design projects over the course of four years. Teams of elementary students with varying expertise worked together for a period of ten weeks designing instructional software to teach about a science topic to younger students in their school. Students moved in their participation from software evaluators to software apprentices and designers emulating different aspects of professional software design practice. We were particularly interested in how students conceptualized their project experience as they moved from newcomers to oldtimers in becoming junior software designers. Moreover, we wanted to know how students saw themselves as software designers within the context of the classroom community and within the larger realm of the computer culture. Interview results showed that students not only perceived the differences in skills but also how these structured their collaborative interactions and responsibilities. Students evaluated their programming skills in relation to others in class and the larger technical world. Our discussion addresses changes in students' participation and their belonging within the computer culture and the applicability of the findings within the larger project.

Keywords:

Introduction

In a recent article "Across the Scales of Time" Jay Lemke (2000) wondered what kind of lasting changes schooling facilitates in the behavior patterns of students: "On what time scales do we imagine that personal identities change significantly? … Their attitudes toward potential careers or value choices?" (p. 282). He observed that most studies focused on relatively minute time periods when considered within the life span of human development. The present paper intends to shed some light on this issue by following the same group of elementary students from third to sixth grade, a period of four years during which they participated in several consecutive collaborative software design projects, a form of project-based learning (Blumenfeld et al., 1991). In these projects, teams of three to five elementary students worked together for a period of ten weeks designing instructional software to teach younger students in their school. Students moved in their participation from software evaluators to software apprentices and designers emulating different aspects of professional software design practice with one important distinction: unlike professional and cognitive models of apprenticeship (Collins, Brown & Duguid, 1992) more experienced peers were in the role of experts, not adults.

In the past we have investigated various aspects of students' participation in this learning environment (Ching, 2001; Ching, Kafai & Marshall, 2000; Kafai, 2002; Kafai & Ching, 2001; Kafai, Ching, & Marshall, 1998; Marshall, 2001). Our following analysis will focus on the development of students' identities and their participation as software designers, a key practice in the collaborative software design projects. Lave and Wenger's (1991) conceptualization of learning as legitimate peripheral participation stresses the importance of the learner's growing identification with the practice and community. We were particularly interested in how students conceptualized their project experience as they moved from newcomers to oldtimers. Moreover, we also wanted to know how students saw themselves as software designers within the context of the classroom community and within the larger realm of the computer culture.

Our focus on students' identification with software design practice ties into the larger debate of who participates within the computer culture (AAUW, 2000; Kirpatrick & Cuban, 1998). From a psychological perspective, researchers such as Turkle (1984) have examined what particular aspects of the computer resonate with programmers' interests and needs. From a sociological perspective, researchers such as Huber and Schofield (1996) have studied interactions and setups of high school computer labs that discourage entry and participation in computer activities; findings which have also been found prevalent on the college level (Margolis & Fischer, 2002). The purpose of our interviews during and after students' participation in the projects was to understand how students reflected on their project experiences and skills in function of their changing project responsibilities and interactions.

METHODS

Participants

The focus group of 12 students interviewed for this study was comprised of 7 boys and 5 girls (Caucasian: 9, African American: 2, Asian: 1). These students were chosen because all of them had participated from fourth until sixth grade in at least five collaborative software design projects; in addition, a subset of them (four students) had been third grade software evaluators thus covering a life span of four years of project experience. The fourth graders who had no previous design project experience and were referred to in student parlance as "first years" whereas in the following year as fifth graders they were referred to as "second years". In sixth grade, students continued the collaborative software design project by working together in teams with other sixth graders new to the collaborative software design project.

Project Activities

The classroom teacher and researchers set up the mixed-grade and -gender teams according to principles of best practice and assigning oldtimers to each team. Prior to starting work on their software projects, students participated in a three session long introduction to MicroWorlds Logo. Each student was responsible for doing research to answer a science research question of the student's choosing relating to the science curriculum topic (e.g., human physiology, marine life) and students had to teach something that they learned in the software. It was up to the teams to decide the content of their software and anything else related to planning or managing their software work. During the first week of the project, each team was given a blank planning board to keep track of their schedules and ideas. In weeks four and eight, third graders visited the classroom to give design teams feedback about their software. At week seven, the teams participated in what the teacher referred to as the "roast": Teams got up in front of the class, and each member explained his or her question and the related software page while classmates were encouraged to critique the questions and software ideas. Science instruction was interwoven throughout the project and consisted of a mix of whole classroom and group activities and discussions. The project concluded with a final software presentation and an evaluation of each team to the rest of the class.

Data Sources

Students were interviewed about their experiences in the collaborative software design projects at the following three time points: T1 when the participating students were newcomers (fourth graders) after having embarked on their first collaborative project experience; T2 when the students were oldtimers (fifth graders) after having completed a second design project; and T3 when the students were in sixth grade and participated in an exit interview reviewing their computer knowledge and project experience. Table 1 provides an overview.

Table 1: Overview of Students' Roles and Interviews at Different Time Points

Time Points	1997 **T0** Third Grade	1998 **T1** Fourth Grade	1999 **T2** Fifth Grade	2000 **T3** Sixth Grade
Project Role	Software Users	Software Apprentices	Software Designers	Software Designers
Interviews		- Q1: Software Designer Identity (Pre/Post) - Q2: Project Experience as Newcomers (Post)	- Q1:Software Designer Identity (Pre/Post) - Q2:Project Experience as Oldtimers (Post)	- Q3:Computer Person Identity - Q4: Review of Project Experience

Interviews. At the end of each collaborative software design project, we interviewed students about their project roles and identities with the following questions:

Q1: Do you think of yourself as a novice software designer? Why or why not?
Q2: What do you think it means to be a first year student working on design projects?
What do you think it means to be a second year student working on design projects?
Q3: Do you see yourself as a computer person? Why or why not?
Q4: Describe what it was like being a third grader when visiting. What did you like best about your visit?
What was it like being a fourth grader on a design project? What do you remember?
What was it like being a fifth grader on a design project? What do you remember?
What things did you like better as a fifth grader than as a fourth grader? What things did you like less?

Data Analysis

All answers to the questions from the interviews were transcribed. The researchers read the interviews independently and the coding schemes were derived (described in more detail in Results section), each researcher then did a blind coding of three samples of each question from three time points and demonstrated a 95% agreement. The 5% of codings upon which there were disagreements were discussed and resolved. After reliability was established, a single researcher then coded all of the remaining data.

RESULTS

Project Experiences as Newcomers and Oldtimers

Our results illustrate that students' awareness of what it means to become a junior software designer became more differentiated as they moved from newcomers to oldtimers. Following a modified scheme developed by Ching (2001) we analyzed how students articulated differences between the roles of first and second year students working on the project (see Table 2). We classified students' answers into four categories: (1) 'unspecified' described general differences with no specific reference to programming or roles; (2) 'knowledge' were answers that pointed out differences in programming skills; (3) 'roles' focused on social aspects in particular how students described their main activity within the team; and (4) 'no difference' was selected when students saw no main distinctions between first and second years.

Table 2: Project Experiences as Newcomers and Oldtimers

	T1 (Newcomer)	**T2 (Oldtimer)**	**Examples of Student Answers:**
	Post	**Post**	
Unspecified	33%	25%	*"The second years have more experience."*
Knowledge	42%	50%	*"The second years have had MicroWorlds before and the first years haven't, so the second years know more of the programming."*
Roles	25%	83%	*"As a second year I think you have to help the first years. You're sort of like a teacher or a leader."*
No Difference	33%	0%	*"Everybody has their strong points so it doesn't really matter if you're a first year or a second year."*

Project experiences changed most dramatically in the category 'roles'. The following answers given at time points T1 and T2 illustrate a student's shift from 'no difference' as a newcomer to 'roles' as an oldtimer:

T1: Madeline as a Newcomer
Madeline: *Well it was kinda strange because I was the only first year in my group so um, not really because like, everybody has their strong points so it doesn't really matter if you're a first year or a second year.*
T2: Madeline as a Oldtimer
Madeline: *Well, um, I - from personal experience – know that it's not as easy as it looks because it's kinda hard teaching the fourth graders.*
Researcher: What do you mean?

Madeline: *Because, I mean, it's not their fault at all, it's just how things are. But sometimes it's hard because you know how to do it, but it's hard to explain it. Like, explain why it is or help them remember and stuff like that. So it's, I mean, it's easy because you know it, like the programming wise, but it's not easy with the fourth graders because you have to take on, like two roles, kind of.*
Researcher: So what are those roles?
Madeline: *Well, a teacher, and like, a programmer.*

Competence in Software Design: Views of Self and Community

When students were asked whether they considered themselves novice software designers we found the following results (see Table 3). What we observe here is a decline in perceived competence: as newcomers after having participated in a first design project 82% of students saw themselves as novice software designers whereas only 55% did so after having participated in four design projects.

TABLE 3: "Do you see yourself as a novice software designer?"

	T1 (Newcomer)		**T2 (Oldtimer)**	
	Pre	**Post**	**Pre**	**Post**
Responses yes	100%	82%	75%	55%
no	0%	18%	25%	45%

In giving their rationale for why they did or did not consider themselves novice software designers, students' answers fell into two categories: (1) references to self and (2) references to others. The following excerpts are examples of students' answers:

Reference to Self
Own Trajectory: specific reference to one's own development in terms of prograr
e.g., "In the beginning I didn't know anything, but now I know more stuff
General Assessment: not very specific
e.g., "I'd say I'm about medium." or "I know a lot of programming."
Programming Knowledge: specific reference to the types programming knowled
e.g., "I know how to program a slider."
Reference to Others
Inside the classroom: comparison of oneself to other students in the classroom
e.g., "I'm okay, but I'm not like Madeleine or Ian." or "The people in my g than me."
Outside the classroom: comparison of oneself to programmers outside of the clas
e.g., "If you compare me to people in this class, I'm pretty good, but outsi not really."
Programming Knowledge: reference to larger world of programming outside of t
e.g., "There is so much more Logo to know, and all I know is some of Mic

Students' rationales changed mostly in reference to others: as newcomers most students do not use reference to others (92%) whereas as oldtimers more than half of the students (55%) appeal to others in their comparisons (see Table 4).

Table 4: "Do you see yourself as a novice software designer? Why or why not?"

	T1 (Newcomer)		**T2 (Oldtimer)**	
	Pre	**Post**	**Pre**	**Post**
Ref. to Others yes	8%	33%	33%	45%
no	92%	67%	67%	55%

From Users to Designers: Views of Changing Participation in Classroom Community

In our exit interviews conducted at time point T3, at the end of sixth grade, we asked students to review their project experiences as software users, apprentices and designers. For those four students who had been software users as third graders, students mentioned aspects such as "I wanted to know what fourth and fifth grade would be like. So I came up and I saw the projects and I'm like 'oh no, I'm gonna have such a hard time to do it'. Because you know on the procedures page, there's all this writing. And so he [fifth grader] showed me the procedures page. I was just confused." (David) or "It was really confusing because all these people were like fifth graders and I was a third grader and they had really, like, tough subjects and I just looked at their animations. It was really new. I thought like, 'man, I want to do that when I get into fourth grade'…I got a little glimpse of what I would be doing in the next two years." (Sheila)

When students then moved into fourth grade as software apprentices, all students (n=12) mentioned the knowledge of the programming language as the most prominent aspect of their experience being software apprentices "when I didn't know how to work Logo, it was kind of hard." (Amelia), followed by social differences "I had to sort of learn on my own and from other people" (Ian), and affective aspects such as "but once I learned it, then I realized it was really fun to work with [Logo]" (Kelly). When students reviewed their fifth grade experience, their increase in knowledge of the programming language was mentioned as frequently as their social interactions when they were able teach other people to program (92%) "It was better because I kind of show people because I was the one who knew what to do. Instead of just being the one who was taught, I could teach people" (Jared). Whereas affective aspects were only mentioned by two students such as "[b]ut the fourth graders kind of have more needs. They're still kind of beginners in this huge learning process than when I was a fourth grader I didn't even get. So I could feel exactly what they were feeling. But also one of the good things about the higher expectations is once you meet those expectations, you're kind of a full [sic] fifth grader" (Anton). Their own comparison of their fourth grade with their fifth grade experience indicated that students liked best being knowledgeable, helping and teaching others whereas their increased responsibilities in terms of keeping the team on task and on schedule and group management issues were listed as the most demanding aspects.

When we asked students whether they saw themselves as a computer person, six of them (50%) answered positively, two said maybe (17%), and four declined (33%), yet eleven of them (92%) considered themselves to be comfortable with computers. Our analysis of rationales indicated that their knowledge of programming was explicitly mentioned in only two instances whereas seven (58%) of them mentioned their knowledge of other applications such as playing games and surfing the Internet; only one person compared himself to other students. A more interesting distinction about being a computer person was drawn by students themselves in statements such as "I love computers. I find them so interesting how they work. And I just find it so amazing that this little box can create all the things." (David). But students also distanced themselves by saying "I wouldn't say I love them [computers]. Like I have friends who like love computers and like can't stand to be not around them and stuff. I'm not like attracted to them." (Kelly) or "No, I'm the more outgoing type of person. I don't like to stay at home and do stuff on the computer. My brother, he's like a computer person" (Amon). These kind of statements reflected the beginnings of an outsider/insider distinction.

DISCUSSION

The goal of this study was to examine how students described their project experiences and developments within the particular culture of the collaborative software design project. Students entered this community as software users evaluating software designed for them by older students in their school. They then proceeded as fourth graders to become software design apprentices working together with more experienced software designers, the fifth graders, on the project. The following year students themselves moved into the roles of software designers or oldtimers apprenticing the new cohort of fourth graders, a role that they repeated as sixth graders. While the collaborative software design projects encompassed by no means the students' whole school experience, a significant amount of time was spent by students in these projects and more importantly, in a practice that involved them in more complex forms of participation over the course of four years. Students' reviews of their project roles and skills echoed this development as did our assessment of their skills. In the following sections we will discuss in more detail how students' views of their participation and skills changed within the context of the collaborative software design project and within the computer culture at large. We will also address the validity of our findings coming from a small group of students by placing them in relationship to findings of our companion research.

Inside Classroom Computer Culture

Wenger's (1998) idea of community membership contributing to one's identity serves as a useful framework for interpreting these results. In his view, one's membership, belongingness, and participation within a community all contribute to the ways in which identity is developed over time, and that the degree to which an individual engages themselves in a community contributes to the development of their identity. We can see in the present study that membership to the classroom community was enhanced through social interactions. Increased participation within the community, as demonstrated through helping (each other on projects) mentioned by many students, provided a growth of one's understanding of their own identity, as demonstrated in the responses to the question on project experiences.

We also noticed a shift in students' understanding of their project roles because in the retrospective review, students no longer focus solely on the importance of programming knowledge. Instead, they also bring up the importance of accountability (for example, making sure work gets done and the group is managed) and the responsibility of providing programming support. The aspect of 'helping others' was mentioned often as a 'distinguishing' feature of being an oldtimer. This accountability aspect is in line with research describing what it means to move from the periphery to more full forms of participation. What is more surprising is the frequency with which accountability is mentioned as the "worst" aspect of being an oldtimer. In Ching's (2001) interviews of students' perceptions of what it means to be a newcomer or an oldtimer, the aspect of planning was virtually absent from students' considerations. It is possible that students did not mention it because it was a negative or difficult experience. It is also possible that these "intangible aspects" of project learning do not get articulated because only after several project experiences (and not just the first time of being an oldtimer) they start appearing as an important factor right next to programming knowledge and skills.

Outside Classroom Computer Culture

If our findings about students' project experience were largely positive, a much more complex picture emerges around the identification with the larger computer culture. Describing oneself as a junior software designer positioned students within the classroom culture. Students' growing understanding moved from using themselves as a reference point to using others. This finding seems to argue that students' understanding of the larger world, computer culture, is starting to expand. Such a view is in line with children's growing participation of the world. A supporting finding is that students' descriptions as computer persons reflects these results. We found that students' identification with the computer culture starts to divide up between insiders and outsiders. Statements such as "I love computers" were mostly pronounced by boys whereas girls were distinct in pointing out that "I like computers but I do not love them". The finding that boys more than girls identify with computers is not surprising. The rationales provided by students such as "I'm more of the social type" or "I like to do other things" seem to indicate that students have started to associate their interest with computers with culturally prevailing stereotypes of computer users as nerds. In fact, these results seem to be in line with a large-scale study that examined children's and adolescents' perceptions of computer users (Barba & Mason, 1994) and found that with the beginning of adolescence the concept of a computer user as a nerd starts to emerge; in elementary school most children associate computer use within a range of professions but not as a type alone. Our findings show that boys as well as girls draw on these distinctions and identify or disidentify with the computer culture at large irrespective of their general proficiencies.

General Applicability

While over one hundred students participated in the collaborative software design projects, we have reported here on the results from twelve interviews. One strength of our findings is their triangulation through different interview questions that we used to approach students' project experiences. For example, the centrality of software design skills was mentioned at all different time points and within the context of different questions; what changed, and this so appropriately, were students' justifications. These changes were indicative of the shift in their roles within the collaborative software design project. Furthermore, when we analyzed the same group of students and their perceptions of helping interactions, we found that these students as newcomers reported mostly about receiving help whereas as oldtimers reported about providing more help to others (Kafai, 2002). We also know from companion research that larger groups of students reflected on the differences in programming skills and teaching roles between newcomers and oldtimers (e.g., Ching, 2001). These complementary findings suggest that the trends observed in the 12 interviews have the potential to speak for a larger group.

IMPLICATIONS

We are turning now to address the larger implications of our findings. It is only in the past ten years that learning researchers have started paying increased attention to issues of identity development and learning. The few available studies have focused mostly on professional settings (e.g., Lave & Wenger, 1991; Hodges, 1998; Wenger, 1998) leaving aside the field of school and younger participants other than adults. Previous reform efforts such as project-based learning have addressed aspects of personal relevance and meaning in the design and examination of learning activities, yet such accounts have an overly individualized stance. A focus on identity development casts a wider net encompassing cultural, emotional and social aspects in conjunction with academic benefits. The present study provided an example of this expanded notion. The students' first person accounts included their own classroom community as well as the larger computer culture as reference points in defining what they know and who they are. It became also clear that not only school work but also home activities and interactions with siblings, relatives, and others play an important role in this development process. What this study offered is a perspective on the multiple forces at play in getting and maintaining students interested and knowledgeable in the digital domain.

Another key contribution from our analysis is that a first participation in the design projects was only the beginning of this process and that over the course of several of these design projects students' notions shifted and expanded to become more refined. While this finding alone shouldn't be surprising, it points to the need to rethink our intervention framework from one study to many to capture this development. In addition, it points to the need to create collaborative developmental structures that allow students to move out of the mostly peripheral forms of participation in traditional school work. What we mean by collaborative developmental structures are versions of students' trajectory from software evaluators to apprentices and designers that are not just repeat performances of the same activity. Such trajectories position students within their teams and their classroom community based on their prior experience. These structures provide a reproductive perspective for students of who they are to become; in other words there is a past and future to their learning.

REFERENCES

American Association of University Women (2000). *Tech Savvy: Education Girls in the New Computer Age*. Washington D.C: American Association of University Women Educational Foundation.

Barba, R. H. & Mason, C. L. (1994). The emergence of the "nerd": An assessment of children's attitudes toward computer technologies. *Journal of Research on Computing in Education, 26*(3), 382-390.

Blumenfeld, P., Soloway, E., Marx, R., Krajcik, J., Guzdial, M., & Palincsar, A. (1991). Motivating project-based learning: Sustaining the doing, supporting the learning. *Educational Psychologist*, *26*, 369-398.

Ching, C. C. (2001). *Apprenticeship, learning, and technology: Children as oldtimers and newcomers in the culture of learning through design*. Unpublished doctoral dissertation, University of California, Los Angeles.

Ching, C. C., Kafai, Y. B., & Marshall, S. (2000). Spaces for change: Gender and technology access in collaborative software design projects. *Journal for Science Education and Technology 9*(1), 45–5.

Collins, A., Brown, J. S. , & Newman, D. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing and mathematics. In L. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Hodges, C. (1998). Participation as dis-identification with/in a community of practice. *Mind, Culture, and Activity, 5*(4), 272-290.

Huber, B. R., & Schofield, J. W. (1998). I Like Computers, But Many Girls Don't: Gender and the Sociocultural Context of Computing. In H. Bromley & M. Apple, (Eds.), *Education/Technology/Power: Educational Computing as a Social Practice* (pp. 103-237). Hew York: State University of New York Press.

Kafai, Y. B. (2002). Elementary students' perceptions of social networks: Development, experience and equity in collaborative software design activities. *Proceedings of the Fourth Conference on Computer Supported Collaborative Learning*, Coulder, CO.

Kafai, Y. B., & Ching, C. C. (2001). Affordances of collaborative software design planning for elementary students' science talk. *The Journal of the Learning Sciences*, *10*(3), 323-363.

Kafai, Y. B., Ching, C. C., & Marshall, S. (1998). Children as designers of educational multimedia software. *Computers & Education, 29*(2/3), 117-126.

Kirkpatrick, H., & Cuban, L. (1998). Should We Be Worried? What the Research Says About Gender Differences in Access, Use, Attitudes, and Achievement with Computers. *Educational Technology, 33*, 56-60.

Lave, J. & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge: Cambridge University Press.

Lemke, J. (2000). Across the scales of time. *Mind, Culture and Activity, 7*(4), 273-290.

Margolis, J. & Fischer, A. (2002). *Unlocking the clubhouse*. Cambridge, MA: MIT Press.
Marshall, S. (2001). *Planning in Context.* Unpublished doctoral dissertation. University of California, Los Angles, CA.
Turkle, S. (1984). *The second self*. New York: Basic Books.
Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity*. Cambridge University Press.

ACKNOWLEDGEMENTS

This research was supported by an Early Career Award (NSF REC-9632695) from the National Science Foundation to the first author. The ideas expressed here do not necessarily reflect the positions of the supporting agency. I would like to thank the teacher, Cathleen Galas, and her elementary students at the UCLA Corinne Seeds Elementary School for their participation in the Learning Science by Design research project over the past five years. Cynthia Ching and Kate Muir participated in conducting the interviews; Kate Muir, Meridith Roberts, Van Phan and Melissa Cantu assisted in transcribing and coding the interviews; Brendesha Tynes and Uliz Gonzales helped with the coding of the exit interviews. Additional thanks to Jane Margolis who helped us with the design of the exit interview and to Susan Jurow who provided further comments.