Entering the Clubhouse: Case Studies of Young Programmers Joining the Scratch Community

Yasmin B. Kafai, Deborah A. Fields*, & William Q. Burke
University of Pennsylvania, *UCLA

Abstract

Previous efforts in end-user development have focused on facilitating the mechanics of learning programming leaving aside social and cultural factors equally important in getting youth engaged in programming. As part of a four-month long ethnographic study, we followed two 12-year old participants as they learned the programming software Scratch and its associated file-sharing site, scratch.mit.edu, in an after-school club and class. In our discussion we focus on the role that agency, membership, and status played in their joining and participating local and online communities of programmers.

With the advent of Web 2.0, the boundaries between consumers and producers of software have become less distinct (Fischer, 2009). Youth are at the forefront of those participating in social networking sites, contributing media content in various forms. More and more youth regularly act as content creators and informal programmers (Lenhart & Madden, 2007). Be it modifying a cell phone ring or creating content for a social website like MySpace, youth employ varying computational skills to tailor media to their own needs and tastes. And while modifying a cell phone ring is hardly the same as writing a compression algorithm, youth are nonetheless regularly encountering and navigating programmable media. To fully participate in Web 2.0 communities requires new competencies of youth (Jenkins, Clinton, Purushotm, Robison, & Weigel, 2006) involving more technical and creative dimensions (Kafai & Fields, 2009; Peppler & Kafai, under review).

While the popularity of file-sharing sites suggests that young end-user designers join and participate in large numbers, in the particular case of programming the situation is often different. Much research has documented over and over again that most youth do not know what programming is, do not have access to programming skills, and perceive it as overly technical and thus not for them (Margolis, 2008). Social and cultural dynamics such as agency, membership, and status are thus instrumental in getting novices into programming (Kelleher & Pausch, 2005; Margolis & Fisher, 2002; Margolis, 2008). The focus of this paper
will be how young programmers negotiate their entry into the club – an online community of programmers – making reference to the landmark research by Margolis and Fisher (2002).

A recently developed file-sharing site for informal programmers, scratch.mit.edu, allows us to examine in more depth how youth become engaged as end-user designers in Web 2.0 in the particular context of Scratch and its associated file-sharing site. As part of a four-month long ethnographic study, we followed two 12-year old participants, Lucetta and Matthew, as they learned the programming software Scratch (Resnick et al., in press) and then joined scratch.mit.edu both in an after-school club and in a class. Participation in Scratch shares many similarities and challenges with professional communities and thus we can study this process among the incoming generation of users.

Background

Traditionally end-user development has been concerned with professionals and how they can customize tools to accomplish their work. Much of the research has either studied what problems end-user designers encounter in this process or how to design tools that would be supportive of their endeavors (e.g., Lieberman, Paterno & Wulf, 2006). This research has been largely separate from efforts that have covered the same territory, albeit with young end-users, in school contexts. Here end-user development has been concerned with designing environments and tools that support novices in learning of programming (Guzdial, 2004). Ultimately, end-user development for professionals was seen as facilitating the modification of tools, whereas the focus for youth was on designing tools for ease-of-use, taking into account the differences in motivation, background, and developing expertise between young learners and adult professionals (Soloway, Guzdial, & Hay, 1994).

The research on learning programming and designing novice programming languages and environments has a long tradition (for an extensive overview see Kelleher & Pausch, 2005). Early studies focused mostly on understanding the design and ease-of-use of specific programming concepts such as loops, conditionals, or recursion (e.g., Soloway & Spohrer, 1989). Several efforts concentrated on designing scaffolds for beginning programmers to ease syntax and control problems (Guzdial, 1995; Jackson, Krajcik, & Soloway 1998) or have developed new genres of interfaces that generated scripts based on users’ interactions (e.g., Cypher, 1993). Perhaps the most longstanding effort has been design of programming languages for students based on an
object-oriented programming paradigm which has now has become an industry standard: Logo and Smalltalk were the predecessors of today’s Agentsheets, Alice, and Scratch (Guzdial, 2004).

When Kelleher and Pausch (2005) reviewed the development of programming environments and languages for novice programmers, they found that most efforts aimed at making the mechanics of programming more manageable. But more importantly, Kelleher and Pausch identified “the lack of a social context for programming, and the lack of compelling context in which to learn programming” (p. 132) as key impediments to getting programming novices involved and supported. They argued that these social and cultural barriers are “harder to address than mechanical ones because they are harder to identify and some cannot be addressed through programming systems.” As the most recent designed novice programming language, Scratch facilitates many mechanical aspects of programming (see description below). In particular media-manipulation features provide the compelling context appealing to youth’s interests in digital media, but Scratch also features a networked community and file-sharing site (scratch.mit.edu) – much like many professional communities which can be accessed by end-users for assistance, feedback, and sharing work (Resnick et al., in press).

In previous research, we examined the introduction of Scratch in a local context of a community technology center and identified the roles of mentors (Kafai, Desai, Peppler, Chiu, & Moya, 2008), ease of media manipulation (Peppler & Kafai, 2007), and community organization support as key factors in getting youth interested in programming (Kafai, Peppler, & Chiu, 2007). We also documented that learning programming is possible in an informal context that lacks the direct instruction and support often present in classrooms or professional training workshops (Maloney, Peppler, Kafai, Resnick & Rusk, 2008). Now that an online file-sharing site extends the community of programmers beyond a local setting, we wanted to examine what it takes to enter and participate as programmers in the Scratch programming online community.

More specifically, we focus on how young software designers develop personal agency with programming, move toward membership in a programming community, and gain status as experts amongst their peers. These three aspects emphasize an individual’s sense of self in creating things (Black, 2006; Hull & Katz, 2006), centralized participation amongst a group of people (Lave & Wenger, 1991), and how individuals are recognized by their peers (Gee, 2000/1). In other words, we are studying how new Scratch users express themselves in their programs, how they begin to participate in an online community of Scratch programmers, and how they began to establish their own statuses.
as expert programmers. Incorporating Scratch in both an after-school and classroom setting provided us with a unique opportunity to view how young end-user designers interacted with programmable technology in two spaces with different affordances in terms of support.

Setting, Participants, Tools, & Methods

Settings. This was the first time we had introduced Scratch into a school setting, having up until this time used the program largely at youth-oriented technology clubs, such as the Computer Clubhouses (Kafai, Peppler, & Chapman, 2009). For four months in 2008, we were engaged in ethnographic research at a metropolitan laboratory school located in Southern California. From February to March, we started with an after-school Scratch Club, an optional extracurricular program that met two to three times a week for an hour at a time. In April and May, we continued during regular school hours, setting up a Scratch class where kids worked in groups of two or three during a set of six hour-long math classes. In both the club and the class members participated in the media-sharing Scratch website, which forms a third setting to our research.

Case Studies. A total of 47 middle-school youth, ages 10-12, participated in the study and were representative of the school’s diverse population of African-American, Caucasian, Asian, Latino, and Middle Eastern descent students. Our analyses here center upon two of the four sixth graders who used Scratch across both the club and the classroom settings. The first, Lucetta, was twelve years old at the time of the study. About 5’5” in height, and of African American and Caucasian descent, Lucetta was quiet at times, jubilant at other times, and always inquisitive. She was one of the first and most regular Club members at the school. The second case study centers upon Matthew, a later arrival to the Club. Eleven years of age at that time and also of African-American descent, Matthew had learned about Scratch from his group of anime- and videogame-loving friends, and was excited to pursue it in both the club and classroom settings. Each of these two users’ distinct personalities were reflected by their divergent interests in the program and their techniques for operating the Scratch technology, and over the course of using the program, each subsequently developed their own unique status as an expert.

Programming Environment and Site. In simple terms, Scratch can be described as a visual programming environment to effectively create games, animations, art, and other interactive media. Scratch allows
designers to create games, animations, art and aesthetics, sound design, and stories (Resnick, Kafai, & Maeda, 2003; Resnick & Silverman, 2005) by manipulating media through a process of dragging-and-dropping command blocks of code then stacking these blocks together to form coding scripts (see Figure 1). On the far left side of the Scratch screen is the series of nearly 90 programming command blocks, allowing the user to manipulate sound, images, motion, and other input. In the lower right side of the screen, there is a cache of Sprites, which can be any imported or hand-drawn characters or objects in the video game. The middle panel represents the particular command blocks that the user has selected and stacked. Once they are double-clicked with the mouse, these stacked scripts activate various selected images and sounds on the Scratch Stage (in Figure 1, a musical animation featuring ocean-life characters of starfish & sea urchin) creating basic games, animations, and types of geometric art that can grow increasingly complex and nuanced depending upon a user’s ability to stack and coordinate a range of command blocks (Maloney et. al, 2008).

Fig. 1: Screenshot of the Scratch interface with the musical project Ocean Music Box
No small part of Scratch’s initial appeal is based upon the informal ease with which users can begin to creatively play with the software to create games, animations, and art that are representative of their own interests and talents. Simply knowing how to “drag and drop” items with the mouse is enough to get any user started, and the program offers many sample projects for the new user to tinker with in the process of acquainting oneself with the software. Re-appropriating others’ creations on Scratch for one’s own purposes is one of the primary ways new users familiarize themselves with the technology. The name of the program itself, “Scratch”, refers to the remix practice of DJs who would appropriate various songs into a single track by way of “scratching” multiple records.

Like in other Web 2.0 applications, Scratch designers are encouraged to share their projects with each other and build off of each other’s ideas and creations. Since its public launch in May of 2007, the Scratch website (http://scratch.mit.edu) has been the primary means for users to share their work with one another (see Figure 2). With over 320,000 projects shared to date, the Scratch website is a vibrant online community with over 1,000 new projects being uploaded every day.

Fig. 2: Screenshot of the Scratch.mit.edu, the file sharing site
Described as the “YouTube of interactive media,” the Scratch website allows designers to not only upload their creations but also download and remix others’ projects, as well as post comments, “friend” other designers, and start discussion threads (Resnick et al., in press). The goal, of course, is for Scratch users to not only collaborate with one another online but to also find affinity groups through the site based on common interests and design techniques.

**Data Collection.** Ethnographic field notes were recorded daily in the after-school setting to capture the overall activity of interactions using the Scratch software. In addition, two video cameras were used to record student interactions in real time in both the Club and class settings. These were set up to focus on particular groups and individual students over the course of the study, and their results were subsequently activity-logged and analyzed. Third, both in the middle of March and at the end of May, a total of 21 students – including Lucetta and Matthew – were selected by the research team to be briefly interviewed about their individual experiences using the Scratch software. In March, this group of interviewees was small in size and consisted of 6 regular attendees of the Scratch Club. In May, the number was larger – 15 students altogether, each of whom had worked with a partner (or two partners, in one case) within the classroom setting. In each 10-15 minute interview – both in March and in May – these participants were asked what worked, what proved to be a struggle, if they received help from others, whether they would use Scratch again, and how they saw themselves as new programmers.

**Findings**

In the first section, we will focus on Lucetta’s and Matthew’s use of Scratch in the club setting, highlighting each one’s individual agency as they became part of the larger Scratch community online as well as established their styles of programming. Lucetta and Matthew each approached Scratch in a unique way, based on (a) the projects they were interested in creating and (b) their particular style of navigating the software. Their divergent creative interests and navigation styles highlight the multiple ways Scratch can be used by end-user developers with different needs, interests, and comfort levels. In the second section, we turn our attention to the classroom environment and examine how Lucetta and Matthew developed reputations as “experts” in Scratch based on their experience from the technology club. While both Lucetta and
Matthew became widely recognized and respected by their peers as Scratch experts, each youth acted on this status of “expert” differently.

Scratch Club

Lucetta. Scratch initially was introduced to the school through the after-school Scratch club. Meeting three days a week, the club had a slow start with only two kids showing up for its first meeting in early February. Lucetta was one of these first two. Arriving on that first day with her hair in two braids, Lucetta had never used Scratch before but was ready to learn and was not hesitant to ask questions, such as how to delete certain Sprites. At the first club meeting, she began experimenting with the various Sprites, stacking scripts at random before double-clicking upon them to test their effects. Lucetta’s enthusiasm for Scratch added an air of excitement to the club during that first week when membership was still growing. Exploring the variety of audio features available in the Scratch media library, Lucetta finally settled on a playful – if repetitive – song entitled “Eggs”. The goofy and endearing melody added to the club atmosphere that first day and over subsequent sessions, and with its instantly familiar beat, “Eggs” acted as something of a theme song for those six weeks in the club.

It also was clear during that first week that Lucetta, in particular, appeared determined to start and finish a whole project in Scratch before moving on to any others. Working on a project she named “Mr. Wiggles” (see Figure 3), Lucetta created an underwater scene in which the central Sprite was an octopus that moved back and forth to the “Eggs” music. However, she had a hard time coordinating multiple images of the creature to create an even back and forth movement without the octopus turning upside down. While all the club “newbies” were encouraged during this first session to visit the Scratch website and explore other projects to facilitate their own creations, no club member opted to share and go online – seemingly as hesitant to let others view their early software designs as they were to explore the work of others.

This hesitancy to visit the Scratch website continued over the entire first week. While membership to the after-school club had grown to seven participants by the end of the week, new members shared what they were working on with each other but never joined the online community. “Yay, I got it to work!” Lucetta exclaimed of her “Mr. Wiggles” project mid-way through the second session, and it was clear that her persistence with this single project had paid off. Since this was one of the first projects to be completed and it had a jazzy tune that played over and over again, everyone in the club knew about Mr. Wiggles. In fact, two other club members subsequently sampled (to
varying degrees) from Lucetta’s initial creation in their own attempts to orientate themselves with the Scratch software.

Fig. 3: Screenshot of the Scratch interface with Lucetta’s first creation, “Mr. Wiggles”.

Taylor was the first of these two club members. Joining the club at the beginning of its second week, she was introduced to Scratch by Lucetta, who showed Taylor her “Mr. Wiggles” project by way of an introduction to Scratch. At the researcher’s suggestion the two girls then painstakingly took a screenshot of the scripts of “Mr. Wiggles” and Taylor created matching scripts in her first project, replacing the octopus Sprite with that of a hippo moving back and forth on the stage. Taylor added wings to her hippo Sprite and created multiple costumes to give the impression of flapping wings. She also used a different background and different music, and on her second day created a different ending to her project – namely, having the hippo land and settle on a large daisy. While Taylor’s final project ultimately looked quite different than Lucetta’s underwater creation, many of its operating scripts were identical, and the girls’ collaboration helped introduce other club members to the idea that users could take scripts from one project to solve a dilemma in another project. However, despite continued efforts on the part of the club facilitator, members continued to see collaboration
largely in terms of working together in the club; even into the second week of the club, few members had opted to go to the Scratch website to share and download creations online.

While Taylor’s sampling of Lucetta’s “Mr. Wiggles” project was a collaborative effort, another club member, Craig, opted to borrow directly from Lucetta without initially giving her due credit. At the end of week three, Craig entered the club reporting that he had made a project just like “Mr. Wiggles” and uploaded it to the Scratch website. While Craig said he had created the project by himself at home, it was an obvious copy – using the same Sprite, having it go back and forth, and utilizing an ocean background. The scripts were not all identical (as in Taylor’s project) because Craig did not have a screen shot of the commands from which to copy, but the entire concept was a blatant and purposeful imitation. However the other club members quickly called Craig out on his slight. "You’re making me feel guilty!" Craig protested, but he subsequently credited Lucetta under the project notes section. Lucetta, for her part, did not appear upset. 'It’s really a compliment that you copied my project,’ she informed Craig from across the room. At this, Craig protested that it was not meant as a compliment, but Lucetta had made her point and the rest of the club was there to witness it. Interestingly, despite the momentary friction it caused, this incident was a major step in getting members to access the Scratch website, with three youth creating Scratch website accounts that day and three more club members making website accounts by the start of the fourth week (for more on sampling others’ creations work to foster collaboration, see Perkel 2008).

Over the next few weeks club members gradually began using more and more features of the Scratch website. At first they only used the site to socialize and to browse: uploading their projects, “friending” each other, working on their website images, commenting on each others’ projects, and browsing projects listed on the main homepage. Then members began to do more focused browsing based on their personal interests – looking for different games, anime movies, or solutions to challenges they faced in their own projects. It was during the fifth week that yet another shift from socializing to downloading and remixing occurred. For instance, Ben wanted to create a laser effect on a gun in the game he had been working on for a couple weeks. He went to the Scratch website and found a game that had a similar effect, downloaded it, took a screenshot of the commands pertaining to the laser and used those in his project. Other club members browsed projects, downloaded ones they liked, and made changes to them they thought would be interesting.
The incident with Craig also helped secure Lucetta’s reputation as a knowledgeable Scratch user among her peers. As a member of the club since its inception, Lucetta was instrumental in raising enthusiasm for the program at the school. Yet despite the enthusiasm she helped to instill, Lucetta seemed to consider herself anything but dynamic. “I’m just slow,” she mentioned during week three, “always the last one to go.” This sentiment appeared to be shared by her mother, who, picking up Lucetta up after one club session, remarked that she was not surprised to learn of her daughter’s patience with Scratch. Being slow, she continued, her daughter always worked very carefully. This carefulness very much was evident in the trial-and-error methodology Lucetta chose to employ in the creation of her “Mr. Wiggles” project. While she clearly enjoyed working with her peers, she also maintained a certain focus on whatever project she currently undertook, tinkering with the scripts and Sprites until she achieved the desired effect. Such focus earned her the respect of her peers in the club. While some kids quickly downloaded already created projects, tinkered with them briefly, and then moved onto another sample project, Lucetta consistently worked bottom-up with each of her projects until she was able to get them “just right.” Though she sometimes used images she found online to make Sprites and participated on the Scratch website by uploading projects and commenting on others’ projects, Lucetta did not remix projects from the Scratch site. Nor did she leave any projects unfinished, as some members did when they got excited about a new idea. Instead her style was to come up with an idea and work it all the way through. Often after she finished one idea she would come up with another idea to add on to the project. Lucetta’s expertise was evident in the fact that she could explain exactly why she chose each coding brick and how they fit together. This know-how was a primary factor leading toward Lucetta’s status as an “expert” among her peers. Consequently, as evident in Taylor’s and Craig’s samplings, the “Mr. Wiggles” animation acted as a touchstone for other club members to work off of as they attempted to navigate the Scratch software.

Matthew. Unlike Lucetta, Matthew arrived to the Scratch club somewhat knowledgeable of Scratch because of some friends who shared an interest in the program as well as anime and video games. Matthew was taller than the rest of his schoolmates, and he often seemed a bit awkward with his lanky arms and legs. However, upon first entering the club during its second week of operation in mid-February, Matthew had a swagger to his walk and exhibited a distinct “I-know-what-I’m-doing” attitude. While the majority of his fellow members started their club experience by sampling Sprites largely from Scratch’s own built-in media library,
Matthew immediately went outside of the program and began Googling sprites from his favorite anime shows on the Internet that he then attempted to import into Scratch. When the club facilitator remarked on his seeming familiarity with Scratch and its capacity to import other forms of digital media, Matthew appeared nonchalant, “Aw, it’s just a big thing. It didn’t come out right.” Despite this initial – and perhaps feigned – nonchalance, Matthew’s excitement for Scratch matched that of Lucetta though their style of programming projects was quite different.

The shift in the club to using the Scratch website had a dramatic effect on Matthew’s use of Scratch. Like many of his peers – especially Craig – he initially was bent on creating video games with the software, but upon discovering a ninja animation on the Scratch website that was based on his favorite anime series, *Naruto*, Matthew focused his efforts on building upon the animation’s storyline. In fact, the Scratch website enabled Matthew to find others who shared his interests in anime and built that into their projects. Whereas Lucetta acclimated herself to Scratch using a trail-and-error approach and initially avoided going online for ideas and support, Matthew more quickly embraced Scratch’s capacity to share content, sampling not only from others’ Scratch projects but from other sources of digital media as well, such as iTunes and YouTube, from which he recorded music onto Scratch. In contrast to Lucetta’s working simpler projects through start to finish, Matthew built upon an extremely complex project, even though he never completed his ambitious undertaking.

Entitled “Ninja Showdown 2”, the animation Matthew remixed was a highly complex mini-movie depicting multiple pairs of combatant ninjas, recounting important events and characters in the storyline of *Naruto* as the ninjas battled back and forth leaping and using special powers (see Figure 4). It was a very challenging project that contained over 40 Sprites and 150 scripts. Impressed by the competing figures, Matthew initially grew interested in changing the background music then decided there was an important ninja battle missing from the movie. For the next several weeks he worked on the scene, building off of the existing sprites.

By the end of the club, Matthew had made significant inroads to creating an additional scene to the Ninja Showdown, tailored in an alternative song based on music he recorded from YouTube, and re-titled the project as “Ninja Showdown II Remix”, and added his username to the credits at the beginning of the movie. While he participated in the Scratch website by browsing, downloading, and remixing projects as well as by “friending” creators whose projects he liked, Matthew was never confident enough to post anything on the Scratch website, feeling there
was still much work to be done before he was ready to share online. This is probably partly a result of choosing to do a very ambitious project and wanting it to be complete before posting it. Nevertheless, in the post-club interview in mid-March, he expressed a new confidence based on his time in the club. “So I've never really gotten to do computer animation,” he remarked, “so I kind of get like I kind have a feel of what that's like now.”

![Ninja Showdown](image)

**Fig. 4:** Screenshot of the anime “Ninja Showdown”, upon which Matthew developed his own project

**Scratch Class**

While Scratch club came to a close in mid-March, we introduced Scratch in the sixth grade classes at the school just over a month later in late April. This was the first time Scratch had been formally introduced during the school day, and the format was decidedly more structured than the club as students had to make a “geometric art” project for math rather than having free-play with Scratch. Scratch was used by sixth graders – four of whom had been previous club members – though this time there was a total of 47 students over two classes as opposed to the dozen or so kids that were regulars at the after-school club. In the class, students worked in groups of two or three, and they were expected to work on a single geometric art project over the three weeks – a total of six hour-
long classes – set aside for Scratch. While students were encouraged to get up from their seats to roam the room, most groups relied on their immediate partners to generate ideas.

Based on the participants from the club who were now also in the class, it quickly became evident that some kids had considerably more experience using Scratch than others. The more experienced Scratch participants – including Lucetta and Matthew – were considered the “experts”, and those “newbies” first encountering Scratch were encouraged by both the classroom teacher and the accompanying class facilitators to reach out to their more experienced peers as they worked on their projects. This distinction between expert and newbie was present from day one and continued throughout the entire six classes; kids quickly identified and shared with each other which of their peers knew how to do what. While a select number of groups had a mix of experts and newbies, most were entirely new to the program and would often track down a more expert member from across the room to get help in a particular area. Students also participated in a number of ten-minute “gallery walks” in which they walked the classroom, exploring others’ works-in-progress. These gallery walks further illustrated the multiple ways Scratch could be used. Whether it was a football simulation game, a garden animation scene, or a colorful work of interactive geometric art, participants could see what their peers were working on and how they approached the software, as well as make note of which design aspects they themselves may want to incorporate into their own projects.

**Lucetta.** Paired up with another 6th grader named Nikki who was entirely new to Scratch, Lucetta modeled a very inclusive leadership, unobtrusively working with her partner and helping others in the class at the same time. Over the six class sessions, the girls traded control of the computer’s mouse and keyboard quite seamlessly, ensuring each had the opportunity to work the controls of the program without even having to ask for them. While Lucetta already had made a number of Scratch animations during her club days, she was careful to listen to all of Nikki’s ideas, and the two often laughed at their various early proposals. “I got to listen to her ideas,” Lucetta remarked in the post-class interview of her partner Nikki, “and then we got to see which ones worked. That way, we both got what we wanted.” The girls’ final project – an animation involving a man calling himself “supergirl” who flew across the screen followed by two blooming flowers made from triangles and squares – was an amalgamation of both their ideas. “She had come up with the super-girl thing,” Lucetta explained, “I wanted him to actually make the flowers. But then it was too hard, so we just had the flowers come.”
Lucetta was also inclusive of Nikki when her own expertise was called upon by another group. Numerous times over the six classes, Lucetta was called over by a fellow group looking for assistance, and many times Nikki accompanied her as well. Even when Nikki did not come along, Lucetta was careful not to be gone too long, always returning to her partner within a few minutes of being called away – such periodic monitoring of more experienced programmers has been observed in other student software design teams (Ching & Kafai, 2008). While Lucetta clearly relished her status of expert and was always happy to quietly oblige another group, she never referred to herself as an expert and seemed to be happy to be considered just another Scratch user.

It is interesting to note that the club members’ expertise was limited in a couple of ways. For instance, while club members were considered “experts” relative to their classmates, they were largely unfamiliar with the kinds of scripts needed to complete the geometric art project – scripts about drawing with the “pen” or stamping images. By the end of the project, Lucetta commented that these were now her favorite scripts. Additionally, because of Internet limits in the classroom, Lucetta could not always use her increased awareness of Scratch’s capabilities. “Are we allowed to go online and get pictures? Are we allowed to go online?” Lucetta wondered aloud during the first class session in April, to which Nikki stared blankly back at her partner, entirely unaware that incorporating downloads from the internet was a possibility with Scratch.

Matthew. In contrast to Lucetta, Matthew did not originally get along so well with his partners and was not initially treated as an “expert” with Scratch. Partly because of pre-existing personality conflicts, his first partners tended to ignore him. In fact, once when his partners were stuck, they asked the teacher for help and the teacher said, “So did you ask Matthew who’s an expert?” This was the first time Matthew had been called an expert in relation to Scratch, and he quickly embraced it. When his first partners rebuffed him, Matthew moved around the room making suggestions to other groups who were new to Scratch. The next day when his first partners solicited then ignored his Scratch advice Matthew exclaimed, “Why are you looking at me like I'm an idiot?! I'm the expert!” Other groups in the class backed up this formal naming of Matthew as expert and one group suggested, “(H)e really should be in our group because he’s helping us a lot and he’s really good at Scratch.” By the end of the second class, it was clear this new group was a much better fit for Matthew, as they nurtured his expert status instead of resenting it, though it still took time for them to integrate Matthew into their work.
Over the remaining four classes, Matthew built on his prior experience in the Scratch club to help guide his partners’ use of the technology, and the trio ended up creating an animation involving an exploding car Sprite which was one of the most intricate and well-received projects created in the class. Matthew’s partners teased out his expertise, bouncing their ideas off of him. They would try out a potential script, while Matthew would pick out any errors in their thought process. Matthew was clearly pleased with their progress – the bravado he initially flashed that second day in the class was now vindicated by his far more subdued role as mentor to his two partners. Uploading his group’s final project to the Scratch website at the close of the classroom sessions, Matthew appeared to have attained not only a newfound confidence with the Scratch software but also with himself as a leader. Indeed, in his post-interview Matthew said that one of the parts of the project he liked best was that, “It kind of felt like leadership skills – because I can naturally use the computer.” So his relative expertise in Scratch also helped him to think of himself as a leader with his peers.

Discussion

Our study set out to examine a new group of end-user designers – young programmers – and how they negotiate their entry into a local as well as a global online community of software designers – and to identify how agency, membership, and status amongst the young programmers affect their participation. Collaborative programming activities are increasingly important in professional communities (Wulf, 1999). While professional end-user designers obviously work under different constraints what concerns the focus of their design tasks and tools, participation in online networks for sharing artifacts and ideas is of relevance in both youth and professional communities. Agency certainly played a key role in club members’ identification with Scratch, an attribute that may very well develop more naturally within an informal, non-prescriptive learning environment. Designed specifically for such informal learning environments (Kafai, Peppler, & Chiu, 2007), Scratch’s success with young end-users is not simply making the technical aspects more manageable but closely correlates with the software’s ability to allow users to engage with the program on their own terms. To an extent, Scratch’s informality in social practice replicates the ubiquitous informal programming environments that youth encounter daily and know so well, be it personalizing their cell phones or updating their statuses on MySpace.
The cases of Lucetta and Matthew demonstrate this in the different kinds of programs they made. Lucetta gravitated toward cute musical projects that often used animals in them, coinciding with her interests in animals at the time; Matthew was drawn toward video games and anime, embracing the opportunity to make his own animation based on his favorite anime series. Even in the class, where there was a more structured project, the groups were able to integrate their interests or styles into Scratch through football, music video, or abstract art themes. Participants thrived on the feeling that they could make anything they wanted and incorporated their various outside interests into their Scratch projects. As an extension of this, they became excited to participate in a larger web community where they found others with similar interests.

The web community furthered both Lucetta and Matthew’s membership in a programming community in different ways. Lucetta friended other users, commented on projects, and uploaded her own projects, taking advantage of the social community on the site. This fit her cooperative social style, mixing with others while sharing an interest in Scratch. In contrast, Matthew embraced the potential of remixing, though there were other aspects of participation that he did not take up such as sharing his own project for validation and feedback from the community.

However, Lucetta’s initial resistance to migrating online and Matthew’s own reluctance to upload his club project to the Scratch website, also suggest to us that establishing membership in a larger programming community is not as easily achieved as we had hoped. To return to the social barriers identified by Kelleher and Pausch (2005), when it comes to introducing programming to novice users, establishing a meaningful and comfortable social context in which to learn programming can be the most significant hurdle. While in both the club and class settings, youth freely and easily interacted with each other and then eventually took ample advantage of Scratch’s online community, most of them seemed to need an initial level of comfort with the online environment before participating in the site. And even after club members began using the site frequently and young programmers began remixing others’ work via the Scratch website, other youngsters remained much more cautious about the remixing process and stayed on the sideline while their peers actively explored it. Such hesitancy to plunge into online communities is clearly a topic that needs further investigation. Too often the popular media paints a picture of all youth actively participating in a wealth of virtual communities with abandon, yet in the cases of Lucetta and Matthew, we witnessed some real hesitancy to engage in the online community for various reasons, and these reasons—be it the anonymity of such sites or a sense that one is not “expert”
enough to participate—warrant further analyses. Simply surfing through the discussion forums of a number of other social and gaming websites also targeting youth audiences, it quickly seems evident that only a small minority of users actually run and regularly populate these communities. Meanwhile a significant majority simply looks on and doesn’t contribute.

Finally, in regard to status, the migration of Scratch from the club to the class enabled Lucetta and Matthew to stand out as experts in programming. Our reference to Lucetta and Matthew as “experts” is of course defined in relation to their classmates’ understanding and experiences with Scratch. By no means do we mean to imply that their proficiency in Scratch programming is comprehensive at this point; rather it is intended to denote their relative familiarity or fluency with the Scratch environment. As in previous research we see here a form of an informal “peer pedagogy” (Ching & Kafai, 2008) emerging that showcases their ability to support and monitor their less experienced peers in Scratch design. More importantly, it provides an indication that the level of expertise required to help others can be quite fluid. In our particular case, a few months of design work with Scratch and a few projects positioned Lucetta and Matthew to provide valuable assistance to their team members as well as the larger class. By extension, this suggests that assistance in end-user design can come from more advanced users as well as experts.

Over time at both the Scratch club and classroom, we witnessed a great deal of informal practices that seem to have migrated into young designers’ activities. For instance, the prominent remixing and repurposing, especially in the club, could conceivably lead to some clashes with school culture that still values individual work overall. In fact, we did see ambivalent attitudes of some club members such as Craig towards crediting the original designs of their adaptations (see also Perkel, 2008). By the same token, the approach of searching out completed scripts to be adopted into their own programs mirrors the way many programmers now search online for existing applets as a starting point for their programs. One might bemoan the fact that young designers no longer start from scratch with Scratch and thus might miss out on important learning opportunities. We, on the other hand, see this as a promising practice for young end-user designers that facilitates their entry while at the same time adopting more “expert strategies”. In fact, these approaches to adopting and repurposing others’ software designs might provide a promising training ground for later professional practice. Experience and expertise of what it means to successfully enter, participate and navigate collaborative portals such as Scratch is something that needs to be nurtured early on.
Acknowledgement

The work reported in this paper was supported by a grant from the National Science Foundation (NSF-0325828) to the first author and an UCLA Research Mentorship to the second author. We wish to thank the coordinators, teachers and students at the Corinne Seeds University Elementary School for their participation.

References


Jenkins, H., Clinton, K., Purushotm, R., Robison, A., & Weigel, M. (2006). *Confronting the challenges of participation culture:*


