Programming & Storytelling: Opportunities for Learning About Coding & Composition

Quinn Burke
Doctoral Candidate
University of Pennsylvania
3700 Walnut St. Philadelphia, PA 19104
burkew@dolphin.upenn.edu

Yasmin B. Kafai
Professor of Learning Sciences
University of Pennsylvania
3700 Walnut St. Philadelphia, PA 19104
kafai@gse.upenn.edu

ABSTRACT
The focus of this paper is to investigate how writing computer programs can help children develop their storytelling and creative writing abilities. The process of writing a program—coding—has long been considered only in terms of computer science, but such coding is also reflective of the imaginative and narrative elements of fiction writing workshops. Writing to program can also serve as programming to write, in which a child learns the importance of sequence, structure, and clarity of expression—three aspects characteristic of effective coding and good storytelling alike. While there have been efforts examining how learning to write code can be facilitated by storytelling, there has been little exploration as to how such creative coding can also be directed to teach students about the narrative and storytelling process. Using the introductory programming language Scratch, this paper explores the potential of having children create their own digital stories with the software and how the narrative structure of these stories offers kids the opportunity to better understand the process of expanding an idea into the arc of a story.

Categories and Subject Descriptors
K.3.2 [Computers and Education]: Computer and Information Science Education – *Literacy*.

General Terms
Human Factors

Key Words
Computers & Writing, Digital Storytelling

INTRODUCTION
In this paper, we take a first step towards understanding how programming activities can support students' narrative and composition skills. While there have been several efforts using introductory programming for storytelling and making games, these focused on facilitating entrance into programming activities but not necessarily on directly supporting kids' capacity to structure and develop a meaningful story. In this paper, we first review the history of creative composition with technology tools, then present a writing and media design tool and our use of it with middle school kids in an after-school program. We conclude by exploring the potential ways coding and creative writing overlap and how this new design tool can tap into students' storytelling abilities to further their aptitude for writing, in general.

IDC 2010, June 9–12, 2010, Barcelona, Spain.

BACKGROUND
When computers first emerged on the K-12 landscape in the early 1980s, their capacity to facilitate kids' composition skills was mostly conceived in terms of their ability to make for "cleaner" looking compositions [4]. No longer burdened by aesthetic concerns, such as bad handwriting, computer word processing allowed students to direct increased attention to more significant considerations such as grammatically revising their papers. Using the computer as a word processor helped minimize aesthetic concerns by creating a standard look to compositions—whether it was 12 or 14 point font, double or single-spaced; kids' compositions with word processor still looked largely identical.

However, in the 1990s, as computers gained the capacity to store and display a growing variety of visual and audio features, the aesthetic look of a paper not only grew more significant but became integral to the text itself. Students' written composition became populated with images and sounds [8], and researchers of new media studies found that integrating visual images with written text could both enhance and accelerate student comprehension [2, 14].

Out of this combination of words, images, and audio (as well as from a broadening conception of "text") came the practice of digital storytelling [13]. With its earliest incarnation in the late 1980s, digital storytelling has since emerged as a growing medium by which to introduce youth to the applications of storytelling, writing, and technology using images and sounds [9, 14]. By accompanying words and multi-media effects, kids have increased opportunities to express themselves creatively and learn to integrate the traditional literacy of writing with those of digital manipulation and media production.

One particular form of digital storytelling that recently has emerged has used programming as a media text production tool. Like new media studies, programming-as-writing relies upon words, images, and sounds to create multi-modal digital stories. But whereas new media studies focused on accompanying words with images, video, and

audio to enhance the text, programming-as-writing treats words as the driving component producing these multimedia features. Instead of co-existing with digital graphics and sound effects on the screen, words act as the fundamental language which animates these graphics and coordinates their movement to sound effects. Bruckman's MOOSE Crossing [1] is an early example of programming-as-writing, allowing players (ages 8-13) to program their own narrative elements in a text-based virtual world. However, whereas inputting code in MOOSE Crossing consisted solely of entering text, more recent programming-as-writing languages such as Alice [5] and Scratch [12] also incorporate graphics into the coding process, adding a visual dimension to the process which helps facilitate entry for first-time programmers.

Papadimitriou [10] names digital storytelling as an ideal way to introduce more youth to programming, using simple narratives to emphasize the utilitarian nature of algorithms, and Kelleher and Pausch's work [7] represents a significant step in using storytelling to this end. Using a modified version of the introductory programming language Alice (appropriately named Storytelling Alice) Kelleher and Pausch found success introducing kids to basic programming concepts such as looping and conditionals through the creation of their own 3D animated stories.

While these studies pairing programming and storytelling offered exciting new ways for youth to "write" their own narratives, they also examined the benefits of storytelling only in terms of programming. Certainly programming is an important skill [6] but using digital storytelling simply as a way to draw kids into programming does not take into account the full and rich ways such storytelling can also be used to develop children's sense of narrative structure and capacity to write more effectively. In addition, digital storytelling within the context of programming could have the double benefit of supporting both the learning of programming (as investigated before) and the learning of writing. It is the latter point that is the focus of this paper.

METHODS

Participants

For six weeks this past Spring (2010), we used the introductory programming language Scratch during an after-school Club in an urban public middle school located in West Philadelphia. The "Storytelling with Scratch Club" consisted of 11 middle-school children, ages 10-14, who met twice a week for an hour after school in the school's computer lab. Participants had learned about the Club through fliers that had been posted in the school as well as circulated among middle-school classrooms. The group was representative of the schools' diverse population of African-American, Caucasian, Asian, Latino, and Middle Eastern students. The goal for participants was to produce their own made-up story in Scratch by the close of the program, which was then shared during an all-class presentation over the last day. While some kids worked directly with each other, all participants regularly shared their developing projects with each other through a series of walks around the room and through less formal day-to-day interactions as kids moved about the room. In total, we collected 11 projects—one from each participant—by the end of the program; these projects as well over ten hours of video and daily field notes served as our sources of data.

Scratch

Like Storytelling Alice, the introductory programming language Scratch allows users to manipulate media to create their own stories Scratch [12] simply through a process of "dragging-and-dropping" command blocks of code, then stacking these blocks together to form coding scripts (see Figure 1). These coding scripts are then activated by various inputs, be it a keystroke or the click of a mouse, bringing to life the various Scratch character "sprites" and backgrounds (in Figure 1, an ocean-life scene featuring a shark and fish sprites). Projects grow increasingly complex and nuanced depending upon a user's ability to stack and coordinate a range of command blocks.

Figure 1: Screenshot of the Scratch project titled "Shark Attack?" & the coding bricks (inset) here determining the yellow fish's behavior in the scene

Storyboards

We made use of storyboards over the second week of the program to help participants organize their ideas and order their narratives into a series of programmable stages. Used as early as 1900 by filmmakers and illustrators, storyboards have more recently entered the classroom, and studies [3, 11] have demonstrated that storyboarding can be an effective tool for primary and middle school children to outline and refine their narratives. Providing kids the option to draw out their ideas (as opposed to simply write them down) offers an alternative pathway into the composition process. Using a pencil, kids drew out these individual shots with the knowledge that these screen-by-screen renderings would not represent the sum total of what their projects would eventually be, but rather would act as a "roadmap" for them as they developed their stories.

FINDINGS

Storytelling as a Gateway into Programming

As with Kelleher and Pausch's [7] use of Storytelling Alice with middle school girls, the participants of the Storytelling with Scratch Club all created their own digital stories

through sequentially-based, object-oriented programming. The "objects" of these programs were the stories' sprites and various background stages, all of which had particular functions and procedures attached to them in order to facilitate a cohesive storyline.

Of the 11 digital stories collected and analyzed, all incorporated multiple characters (minimally, a protagonist as well as an antagonist) and 8 out of 11 (73%) of the digital stories made use of multiple stages, which included a train yard, a castle, a city building, and a field of wheat, among other settings (see Figure 2 below).

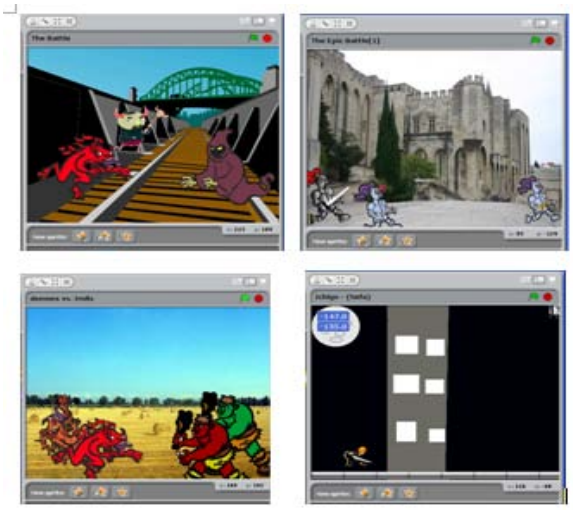
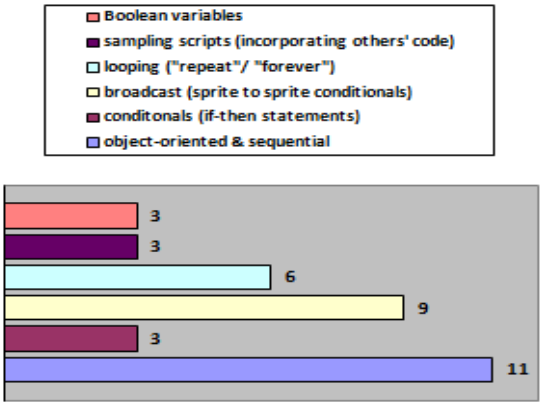


Figure 2: Various screenshots of the Scratch projects collected & the settings they utilized

During those first two weeks of the Club, storytelling proved to be a highly effective way to introduce Scratch to the participants (out of the 11, only 1 child had any significant experience using the program). Rather than focusing on the technical components of the software, we introduced Scratch using the writing terms of "setting", "climax", "antagonist", "protagonist", with which kids were already familiar from the school day. Introducing Scratch sprites in terms of protagonists and antagonists that first day, we were taken aback (but happily so) when a fifth-grade participant raised her hand and exclaimed, "You don't have to keep going over this—we already know what a protagonist is." Our framing of Scratch usage in terms of storytelling gave kids a certain confidence to freely and creatively explore the software over those first two weeks of the Club.

However, over the remaining four weeks, as kids began to expand upon their individual storylines and develop their own digital stories, programming aspects such as debugging scripts and coordinating coding sequences grew incrementally more important to a story's success. Requiring the patience of "guess-and-check" tinkering, these programming issues were far less easy to resolve than the narrative and imaginative questions of weeks one and two. Consequently, over weeks three through six, Club participants quite often hit a wall in the creation of their narratives. Whether it was not understanding why a particular sprite didn't move at the right moment or trying to precisely coordinate the succession of stages as the narrative progressed, kids occasionally grew exasperated by the coding process and some were ready to start a new project altogehter. Yet it was precisely at these times that the elements of storytelling already set in motion became the primary incentive to keep participants engaged with their budding digital creations. Whenever kids hit a wall with their programming, we regularly encouraged them to return to their storyboards for inspiration. And returning to the original roadmap indicating where the story was supposed to go consistently was far greater an incentive for kids to keep debugging codes and generating scripts than simply having us technically pick apart where their scripts had run astray. In the closing questionnaire given to all kids in week six, 7 out of 10 kids reported that the storyboards were useful in the creation of their digital stories, and 9 out of 10 kids indicated that they felt they knew more about programming and computers at the end of the Club.

Table 1: Some of the programming concepts utilized by club members' projects



Programming as Reflection on the Storytelling Process

It has been argued that storytelling is the main mode of successful communication. Whether it's using a parable to teach a lesson or successfully explaining how the sun's rays reach the earth, the ability to frame and pace out a narrative cannot be overestimated as a fundamental skill. So how did Storytelling with Scratch help facilitate kids' storytelling abilities? On the closing questionnaire, 60% of respondents indicated they felt their storytelling abilities were improved over their time in the Club, and of course, while we of would have liked to have seen a 100% response rate to this question, we did see kids storytelling abilities affected in two ways:

First, by introducing and explaining the Scratch software in terms of storytelling, our intent was to underscore the ubiquitous and utilitarian nature of storytelling, allowing it to operate across multiple mediums. Kids programmed sprites for the sake of "character development", multiple Scratch stages were incorporated in order to "rise to a climax", and while "stock characters" behavior could be

frequently "looped", the protagonist's coding scripts usually had to be unique.

Yet many of these same elements of successful storytelling listed above could also be developed by handing kids a video camera—or even a pencil and paper. While traversing multiple mediums certainly highlights storytelling's reach, what, in particular, does programming-as-writing tell us about the storytelling process itself? To answer this—and address the second way kids developed as storytellers—we return to Club participants' regular tinkering with their Scratch scripts, re-adjusting numerical values and re-ordering coding bricks. Programming, kids quickly learned in the Club, is frequently a "guess-and-check" venture, requiring multiple revisions. The same is true of effective storytelling and writing alike; however, while writing and storytelling both typically require a "finished product" before the effectiveness of the narrative can be evaluated, programming-as-writing through software such as Scratch offers a more immediate revisionary process. Kids in the Club continuously ventured and then proceeded to test the effectiveness of each and every one of their potential coding scripts upon the Scratch stage. Did Sprite A meet Sprite B where it was supposed to based on the script? Did the sound effects at the end coordinate with the action sequence? Feedback in the Club was immediate and continuous (and did not rely of us as coordinators) because the direct effects of each coding sequence played immediately out upon the Scratch stage. Thus, for Club members, the process of composing and revising their stories was a synchronous one.

DISCUSSION

"So what, if anything," asks a 2008 Pew Internet and American Life study, "connects the formal writing teens do and the informal e-communication they exchange on digital screens?" Certainly our limited study here has no ready response to this important question, but it does suggest there is a great deal of learning potential in this intersection of the formal writing practices taught during the school-day and the "informal" activities of digital creation which regularly hover around schools as extracurricular activities, and storytelling represents a common ground for both.

Certainly the limited number of kids involved with our study means it only offers a glimpse of the potential of programming languages to reinforce the narrative and composition skills kids are leaning in the classroom. Moving forward, besides increasing the number of participants, we would also like to pair our Club with a language arts classroom within the same school as this would be much more helpful in distinguishing those writing and storytelling skills kids acquire within a formal learning environment from those skills they develop in the less monitored and less structured realms of afterschool and digital learning.

This split between school learning and digital learning also raises the question as to what rubrics would best serve kids in a Club like ours. We evaluated kids' digital stories largely in terms of common language arts considerations such as plot, character development, and resolution. We paid far less attention to the clarity and efficiency of Scratch users' coding scripts. These scripts deserve more attention; our study here only examines what stories these codes enacted. Further examination may very well uncover that certain similarities exist among kids' coding sequences based on the types and genres of the stories they created. The specific selection and sequences of Scratch programming bricks may potentially indicate users' coding is not just functional but also holds the imaginative and aesthetic qualities characteristic of creating fiction.

REFERENCES

1. Bruckman, A. MOOSE crossing: Construction, community, and learning in a Networked Virtual World for Kids. Unpublished PhD Dissertation, MIT, Cambridge, MA. http://www.cc.gatech.edu/~asb/thesis/.
2. Burmark, L. Visual presentations that prompt, flash, & transform. *Media & Methods*, *40*, 6, 4-5.
3. Caldwell, H. & Moore, B.H. The art of writing: Drawing as preparation for narrative writing in the primary grades. *Studies in Art Education*, *31*, 4, 207-219.
4. Cochran-Smith, M. Word processing and writing in elementary classrooms. A critical review of literature. *Review of Educational Research*, *61*, 1, 107-155.
5. Cooper, S., Dann, W., & Pausch, R. Alice: A 3-D tool for introducing programming. *Proceedings of 5th Annual (2000) CCSC Northeastern Conference*. Ramapo, NJ.
6. diSessa, A. *Changing Minds: Computers, Learning, & Literacy*. MIT Press, Cambridge, MA, 2000.
7. Kelleher, C. & Pausch, R. Using storytelling to motivate programming.*Communications of the ACM*, *50*, 7, 59-64.
8. Kress, G. Visual and verbal modes of representation in electronically mediated communication. In Snyder, I. (Ed.), *Taking Literacy into the Electronic Era* (pp. 53-79). Routledge, London, 1999.
9. Long, J. Telling stories with technology. *Technology & Learning*, *28*, 11, 62.
10. Papadimitriou, C. MythematiCS. *ACM SIGSCE Bulletin: Special Issue on the 8th Annual (2003) ITCSE Conference*, *35*, 3, 1-1.
11. Reeder, K. Using storyboarding techniques to identify design opportunities. *The Technology Teacher*, April 2005, 9-11.
12. Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., Kafai, Y. Scratch: Programming for everyone. *Communications of the ACM*, November 2009.
13. Robin, B. Digital storytelling. *Theory Into Practice*, *47*, 220-228.
14. Ware, P.D. & Warschauer, M. Hybrid literacy texts and practices in technology-intensive environments. *International Journal of Education Research*, *43*, 432-445.