Toward a Block-Based Programming Approach to Interactive Storytelling for Upper Elementary Students

Andy Smith[0000-0002-6577-7764]1, Bradford Mott[0000-0003-3303-4699]1, Sandra Taylor[1], Aleata Hubbard-Cheuoua[0000-0002-7310-1326]2, James Minogue[1], Kevin Oliver[0000-0001-8463-0621]1, and Cathy Ringstaff[0000-0002-2856-1120]2

[1] North Carolina State University, Raleigh NC 27695, USA
{pmsmith4,bwmott,smtayl23, jminogu, kmoliver}@ncsu.edu
[2] WestEd, Redwood City CA 94063, USA
{ahubbar, cringst}@wested.org

Abstract. Developing narrative and computational thinking skills is crucial for K-12 student learning. A growing number of K-12 teachers are utilizing digital storytelling, where students create short narratives around a topic, as a means of creating motivating problem-solving activities for a variety of domains, including history and science. At the same time, there is increasing awareness of the need to engage K-12 students in computational thinking, including elementary school students. Given the challenges that the syntax of text-based programming languages poses for even novice university-level learners, block-based programming languages have emerged as an effective tool for introducing computational thinking to elementary-level students. Leveraging the unique affordances of narrative and computational thinking offers significant potential for student learning; however, integrating them presents significant challenges. In this paper, we describe initial work toward solving this problem by introducing an approach to block-based programming for interactive storytelling to engage upper elementary students (ages 9 to 11) in computational thinking and narrative skill development. Leveraging design principles and best practices from prior research on elementary-grade block-based programming and digital storytelling, we propose a set of custom blocks enabling learners to create interactive narratives. We describe both the process used to derive the custom blocks, including their alignment with elements of interactive narrative and with specific computational thinking curricular goals, as well as lessons learned from students interacting with a prototype learning environment utilizing the block-based programming approach.

Keywords: Narrative-centered learning, block-based programming, digital storytelling.

1 Introduction

Digital storytelling, which combines traditional storytelling with rich digital technologies, has emerged as an innovative approach for engaging students in deep, meaningful learning [1–3]. Digital storytelling enables the creation of effective and engaging learning experiences that support diverse individual needs [2] and creative exploration of

scientific phenomena [4]. However, studies have shown that the success of digital storytelling activities can vary between individual students, and can be improved through interventions such as emotional priming [5], focusing the story creation on scene level details rather than macro story structure [6], and by having students enact their stories through a physical medium such as a puppet show [7].

As a process, creating, refining, and presenting a story reflects many computational thinking concepts. There is a growing recognition of the importance of enabling K-12 students to engage with and develop competence in computational thinking [8, 9]. However, teachers at the elementary levels often lack computer science or computational thinking training, raising the need for activities and tools that can help them integrate computational thinking into their classrooms. Building on Papert's [10] work on programmable environments for learning, creating rich block-based programming environments to support novices has seen success in bringing computational thinking to K-12 education [11]. Block-based programming environments remove many of the hurdles to programming activities, such as the need to learn the complex syntax of traditional programming languages. However, difficulties remain, especially in younger grade levels, leading to a growing body of research providing guidelines for creating age-appropriate learning environments to support computational thinking [12].

In this paper, we investigate how to engage upper elementary students in the creation of interactive narratives while simultaneously developing computational thinking practices. This paper describes our initial work in developing a narrative-centered learning environment for computationally-rich digital storytelling using custom blocks designed to facilitate age-appropriate block-based programming. The learning environment and custom blocks are described in detail, as well as results from a pilot study to evaluate the usability and effectiveness of the learning environment.

2 Related Work

Narrative offers an exceptionally promising tool for engaging students in computationally-rich problem solving. Narrative experiences are powerful, helping us connect with others and understand the world around us [13]. Narrative has a unique ability to serve as an effective means for communicating personal understandings of concepts, such as science phenomena, to others [14]. While prominent in elementary education, storytelling is often exercised in the context of language arts, and actualized with either pencil and paper or word processing programs. Digital storytelling seeks to augment the creativity and effectiveness of storytelling, while also expanding storytelling to other subjects [1]. While there are many variants, most digital storytelling activities revolve around students researching a topic, then creating a multimedia presentation of the story that is presented to their teacher and peers [1]. Digital storytelling activities have shown benefits to student learning and engagement [2], visual memory and writing skills [15], and 21st century skills such as problem solving, argumentation, and cooperation [16].

However, the benefits of digital storytelling are at least partially dependent on students' ability to construct narratives, increasing the importance of a well-designed story creation environment, especially for younger learners. One approach to mitigate these

concerns has been to embed the story creation process into a more scaffolded context including block-based programming environments. In the La Playa environment, students from ages 9 to 11 use a modified version of Scratch [11] to create animated stories involving sprites, audio clips, and events triggered by user inputs [17]. Horn, Al-Sulaiman, and Koh [18] utilized a tangible, sticker-based block language to engage students in creating programs in the context of an interactive storybook. While many of the narratives created by these systems focus heavily on sprite manipulation and animation, this work extends these efforts by designing and investigating a block-based programming environment focused primarily on interactive stories by guiding students through the creation of interactive teleplays.

3 INFUSECS Narrative-Centered Learning Environment

To enable upper elementary students to create interactive digital stories, the INFUSECS narrative-centered learning environment was designed and developed. INFUSECS specifically focuses on the creation of interactive teleplays. Teleplays consist of several parts, including multiple scenes, multiple characters, dialog between characters, and narration. An additional feature of the environment is supporting audience participation to allow for interactivity in the teleplays. As students develop their teleplays, they can insert moments in the narrative where they would like to ask the audience a question, and change the story depending on how the audience responds.

For example, imagine a scenario where students are asked to create a narrative on how a shipwrecked crew is rescued from a remote island. A sample teleplay (shown in Figure 1) would describe a set of characters and where on the island the characters were located. After some dialog discussing possible solutions to the crew's predicament, the audience might be asked if the crew should signal for help with a light source using Morse code, or use a siren to create noise to signal a passing ship. Based on the audience response, the narrative can branch and present the rest of the teleplay reflecting the

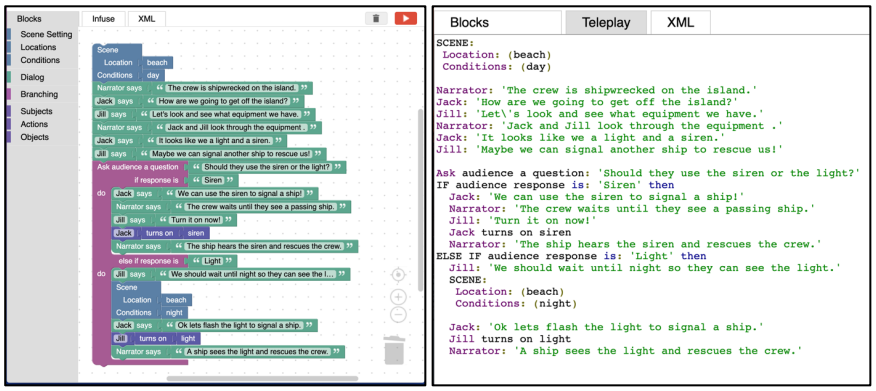

Fig. 1. Screenshots of a sample narrative displayed in INFUSECS's block-based story tab and teleplay tab

audience's choice. Alternatively, the audience participation may only affect a smaller portion of the teleplay, such as allowing the audience to pick which message is sent.

This story creation process engages learners with several core computational thinking practices, as defined by the K-12 Computer Science Framework (www.k12cs.org), including recognizing and defining computational problems, developing and using abstractions, creating computational artifacts, testing and refining computational artifacts, and collaborating around computing.

Teleplays are created in INFUSECS by leveraging Google's Blockly framework [19]. Blockly is an open source JavaScript library for building block-based programming editors. Compared to other block-based programming editors such as Scratch and Snap!, Blockly supports several features desirable for creating a block-based environment for elementary-grade students. First, Blockly allows for a customizable toolbox and block palette. This helps mitigate cognitive load issues by presenting students with only the blocks needed for a given scenario, rather than all the available blocks in the programming language. Second, Blockly facilitates the creation of custom blocks, allowing developers to redesign existing blocks to create the necessary functionality to support digital storytelling. Blockly also allows for the generation of text-based representations of the blocks. While this feature of Blockly is normally used to generate text-based code equivalents of block-based programs, in INFUSECS this feature is used to generate script-like representations of the teleplays, visible at any time via a *Teleplay* tab at the top of the interface.

The INFUSECS learning environment and its custom blocks also leverage findings from previous work with block-based programming environments for upper elementary students [17, 20, 21]. These findings include leveraging visual affordances in the appearance of the blocks, prioritizing sequential programming over event-driven programming, enabling customization and editing rather than being fully generative, and limiting complicated syntax and vocabulary.

The set of custom blocks for INFUSECS was designed to support four main features of interactive narratives: describing a scene, character dialog (including narration), branching, and actions to be performed while performing the teleplay.

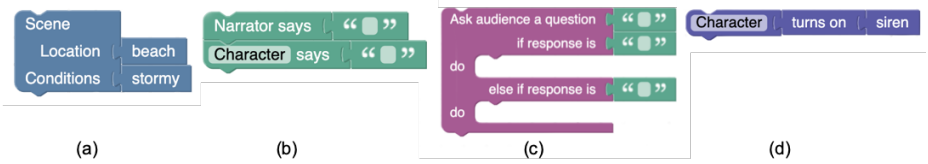

Fig. 2. Custom blocks used in the INFUSECS learning environment

The *Scene* block, shown in Figure 2a, allows students to set the location and conditions where the events in the narrative take place. Location and condition blocks are defined before the activity to fit with a motivating scenario framing a particular digital storytelling activity (e.g., how a shipwrecked crew survives on a remote island). This also allows INFUSECS to leverage Blockly's type-checking capabilities, and prevent students from attaching incompatible blocks to fields, a feature also afforded by the color-coding scheme of blocks.

The *Dialog* blocks, shown in Figure 2b, are the core blocks of the storytelling system, as the majority of the content in the student-generated teleplays is dialog between characters or statements by the narrator. A line of dialog is generated through the combination of a *Character says* block and a *Text* block. The name property of the Character block is editable, and an additional *Narrator says* block is provided.

The *Ask audience* block, Figure 2c, is designed to allow students to incorporate audience participation into their teleplays, while also providing them an opportunity to experiment with conditionals and flow control. The *Ask audience* block contains three properties that must be defined by learners. The first property is the question that will be asked to the audience, and then the two possible responses the audience should respond with to the question. Below each response, students can place the dialog and events that are specific to that branch of the narrative.

Finally, a set of *Action* blocks (example shown in Figure 2d) were added to allow for denoting times when an action may be taken during the performance of the teleplay (i.e., stage directions). *Action* blocks consist of three separate blocks; a subject, an action, and an object. Subjects are the character in the story completing the action. Objects and actions are designed to be pre-populated based on the learning domain in which the storytelling activity is situated. For example, the scenario described in the sample story above is designed to be used as part of a science unit on electricity, thus the blocks included were aligned with items commonly found in science classrooms, and focused on actions like turning on and off a siren or light. These actions are color-coded to differentiate them from other blocks, and can be developed to correspond to items that can be used as props when students are presenting their final story to the class.

4 Pilot Study

To better understand how the INFUSECS learning environment supports digital storytelling and computational thinking in upper elementary classrooms, a pilot study was conducted at a K-8 school in the southeast United States. The primary goal of the study was to observe how pairs of students working together were able to use INFUSECS to create a story as well as to obtain feedback on the successful and unsuccessful aspects of the learning environment and how it could be improved.

Participants in the pilot study included 6 fourth grade students ages 9-10, consisting of 4 males and 2 females. The pilot study took place over approximately 60 minutes. The 6 students were grouped into pairs to work together. Students reported a range of previous experience with block-based coding environments, with 3 students reporting a familiarity of 6-7 out of 10, and 3 students reporting a familiarity of 1-3, with 1 being low familiarity and 10 being high familiarity.

Before using the INFUSECS learning environment, students were given a brief presentation on the learning environment and story development task, as well as a short video introducing a motivating scenario to act as the inciting event for their stories. The video presented how a team of scientists were shipwrecked on an island in the South Pacific after a powerful storm. Before the ship sank, the scientists were able to transport

most of their equipment to shore. Students were then told that they would be creating stories about what the crew would do next and how they would get off the island.

Next, the students were given a brief tutorial on using INFUSECS and began working in dyads to plan their stories for a period of 5 minutes. A brief starter story was loaded into the environment, demonstrating a *Scene* block, a line of dialog from the narrator describing the shipwreck, and an *Ask audience* block to encourage students to include audience participation in their stories. Student pairs then used the learning environment for approximately 30 minutes to author their stories. Each group was paired with a member of the research team, who took notes on their observations, and were available to answer questions the students had about using the learning environment. After creating their stories, all six students gathered together as a group with a member of the research team for a brief 10 minute focus group where they discussed their overall impressions of the learning activity and INFUSECS.

Fig. 3. Students creating a story using the INFUSECS learning environment

All three groups showed an ability to use the learning environment to create short narratives in the allotted time. Each group was able to define the location and conditions of an initial scene, as well as generate dialog between characters. Overall, groups generated stories ranging from 9-12 blocks in length, consisting mostly of narration and dialog between characters. Two of the 3 groups utilized *Action* blocks, and 2 of the 3 groups incorporated a scene change into their narrative. Students also demonstrated a conceptual understanding of the structure of the teleplays, such as editing every occurrence of a character's name in the program and dialog when they decided to remove a character or modify the character's name.

All groups were able to manipulate the blocks to change the ordering of events, often reading back what they had developed aloud, or using the *Teleplay* tab to view their story before returning to the blocks to incorporate revisions. Students also were able to successfully edit and revise their stories, though there were some usability issues such

as text blocks highlighting the entire text when clicked, which caused some dialog to be erased rather than edited as intended. Students were able to utilize the domain-specific *Action* blocks provided, though they struggled incorporating them coherently into their story, highlighting a need to further scaffold this content either beforehand, or during the revision process.

All 3 groups struggled to use the *Ask audience* block effectively. While in discussions with their partners, they exhibited some evidence of understanding how it worked, as well as a desire to utilize the block in their story, only one group was able to create a question with multiple responses, though one of the branches was not fully populated. During the focus group after the students finished using the learning environment, all students were generally positive about INFUSECS. Of interest, they noted that while typing was difficult, they found the process much easier than writing a story with pencil and paper or other tools. One student suggested allowing the recording of audio clips to reduce the typing burden. The main points of confusion expressed by the students were around the *Ask audience* block. Other suggestions for improvements included the addition of more scene locations and weather conditions, and more types of actions and objects for the domain-specific blocks. Additionally, the students requested improvements to the visualization of their teleplays, with suggestions ranging from an animated version of their blocks being generated, to a text version that would play out the blocks sequentially rather than showing the entire teleplay at once.

5 Discussion

Overall, the INFUSECS narrative-centered learning environment was successful in enabling students to create play-like narratives, while engaging students in computational thinking concepts such as abstractions, debugging/revising, order of execution, and conditionals. Though the narratives students developed were short, given more time with the learning environment, higher quality narratives could likely be produced. One set of issues observed by multiple members of the research team during the pilot study revolved around collaboration within the student dyads. Often it appeared that each student was attempting to tell a different story, leading to disagreement and distracting from the goal of the activity. While collaboration is an important practice in K-12 education, more scaffolding or structure is needed to ensure successful collaboration.

A promising potential improvement to the learning environment is to incorporate a longer and more structured planning phase in the activity. One student mentioned that their group was distracted changing character names at times, and that perhaps if they defined the story and characters in more detail before entering the block-based environment, they would have been more productive. While a planning phase could be conducted outside the environment using pencil and paper, incorporating it into the learning environment would enable desirable features such as generating dialog blocks prepopulated with character names to reduce typing, and initializing the workspace with parts of the narrative laid out from the planning phase.

The main point of student confusion related to the block-based programming interface was the *Ask audience* conditional block. Though students generally seemed to understand how the block worked conceptually, only one group was able to successfully utilize it. Potential improvements include providing examples of its usage to students beforehand, and having the block prepopulated with an example question and responses rather than being empty.

There are several limitations to these findings, most notably the small sample size and abbreviated time students had to use the learning environment. However, the system was successful at enabling the creation of narratives, and will be available for use by interested parties.[1] Additionally, the source code of the environment is provided for those that wish to modify and extend the set of blocks available in the learning environment.

6 Conclusions and Future Work

As the importance of engaging learners at all grade levels in computational thinking continues to grow, the need for novel and effective learning activities also increases. Digital storytelling is a natural fit for this need, and a potentially effective way of embedding computational thinking into existing subjects like science and language arts. Conversely, the structured format and intuitive interface of block-based programming can help overcome some of the difficulties students face when constructing stories, allowing them to better focus on communicating their ideas and experimenting with revising their final product.

To better enable digital storytelling in upper elementary classrooms, the INFUSECS narrative-centered learning environment was created leveraging the Blockly framework. A set of custom blocks was designed specifically to facilitate digital storytelling, while also engaging students with important computational thinking concepts. A pilot study was conducted with 6 fourth grade students along with a focus group to gather feedback from the students, demonstrating the usability of the system for students at this age level, and highlighting the promise of the learning environment for enabling the creation of interactive narratives. Overall, the feedback showed students preferred the block-based interface over traditional pencil and paper activities and indicated several areas for improvement.

Future iterations of the learning environment will focus on expanding the set of storytelling blocks available to students, as well as investigating integration of the generated narratives with visualization technologies. Additionally, the environment will be expanded to include a more robust story planning phase and interface to facilitate more structured and higher quality stories.

Acknowledgments. This research was supported by the National Science Foundation through grants DRL-1921495 and DRL-1921503. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

[1] http://projects.intellimedia.ncsu.edu/infusecs/

References

1. Robin, B.R.: Digital storytelling: A powerful technology tool for the 21st century classroom. Theory Into Pract. 47, 220–228 (2008).
2. Smeda, N., Dakich, E., Sharda, N.: The effectiveness of digital storytelling in the classrooms: a comprehensive study. Smart Learn. Environ. 1, 1–21 (2014).
3. Yang, Y.T.C., Wu, W.C.I.: Digital storytelling for enhancing student academic achievement, critical thinking, and learning motivation: A year-long experimental study. Comput. Educ. 59, 339–352 (2012).
4. Henriksen, D., Mishra, P., Fisser, P.: Infusing creativity and technology in 21st century education: A systemic view for change. Educ. Technol. Soc. 19, 27–37 (2016).
5. Rao, N., Chu, S.L., Faris, R.W., Ospina, D.: The effects of interactive emotional priming on storytelling: An exploratory study. In: Proceedings of the International Conference on Interactive Digital Storytelling. pp. 395–404. Springer International Publishing (2019).
6. Brown, S.A., Chu, S.L., Loustau, T.: Embodying cognitive processes in storytelling interfaces for children. In: Proceedings of the International Conference on Interactive Digital Storytelling. pp. 357–363. Springer International Publishing (2019).
7. Chu, S.L., Quek, F., Tanenbaum, J.: Performative authoring: Nurturing storytelling in children through imaginative enactment. In: Proceedings of the International Conference on Interactive Digital Storytelling. pp. 144–155 (2014).
8. Grover, S., Pea, R.: Computational thinking in K-12: A review of the state of the field. Educ. Res. 42, 38–43 (2013).
9. Shute, V.J., Sun, C., Asbell-Clarke, J.: Demystifying computational thinking. Educ. Res. Rev. 22, 142–158 (2017).
10. Papert, S.: Mindstorms: Children, computers and powerful ideas. Basic Books, Inc. (1980).
11. Resnick, M., Maloney, Andrés Monroy-Hernández, J., Rusk, N., Eastmond, E., Brennan, K., Rosenbaum, E., Silver, J., Silverman, B., Kafai, Y.: Scratch: Programming for all. Commun. ACM. 52, 60–67 (2009).
12. Hill, C., Dwyer, H.A., Martinez, T., Harlow, D., Franklin, D.: Floors and flexibility: Designing a programming environment for 4th-6th grade classrooms. In: Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE). pp. 546–551 (2015).
13. Bruner, J.: Acts of meaning. Cambridge, MA: Harvard University Press (1990).
14. Avraamidou, L., Osborne, J.: The role of narrative in communicating science. Int. J. Sci. Educ. 31, 1683–1707 (2009).
15. Sarica, H.Ç., Usluel, Y.K.: The effect of digital storytelling on visual memory and writing skills. Comput. Educ. 94, 298–309 (2016).
16. Niemi, H., Multisilta, J.: Digital storytelling promoting twenty-first century skills and student engagement. Technol. Pedagog. Educ. 25, 451–468 (2016).
17. Franklin, D., Skifstad, G., Rolock, R., Mehrotra, I., Ding, V., Hansen, A., Weintrop, D., Harlow, D.: Using upper-elementary student performance to

understand conceptual sequencing in a blocks-based curriculum. In: Proceedings of the Conference on Integrating Technology into Computer Science Education, ITiCSE. pp. 231–236 (2017).
18. Horn, M.S., AlSulaiman, S., Koh, J.: Translating roberto to omar: Computational literacy, stickerbooks, and cultural forms. In: Proceedings of the 12th International Conference on Interaction Design and Children. pp. 120–127 (2013).
19. Fraser, N.: Google blockly-a visual programming editor, https://developers.google.com/blockly/.
20. Dwyer, H., Hill, C., Hansen, A., Iveland, A., Franklin, D., Harlow, D.: Fourth grade students reading block-based programs: Predictions, visual cues, and affordances. In: Proceedings of the 2015 ACM Conference on International Computing Education Research (ICER). pp. 111–120 (2015).
21. Weintrop, D., Hansen, A.K., Harlow, D.B., Franklin, D.: Starting from scratch: Outcomes of early computer science learning experiences implications for what comes next. In: Proceedings of the 2018 ACM Conference on International Computing Education Research (ICER). pp. 142–150 (2018).