

## Computing for all?: Examining critical biases in computational tools for learning

**Breanne K. Litts, Kristin A. Searle, Bryan M. J. Brayboy and Yasmin B. Kafai**

*Breanne K. Litts is an Assistant Professor in Instructional Technologies and Learning Sciences and director of Learn Explore Design Lab at Utah State University. Together with Indigenous communities, formal and out-of-school educators, and other community organizations, she examines how young people construct their identities through place and story, how to use technology to bring people together in cross-cultural contexts, and how people collaborate across disciplines, communities and cultures. Kristin A. Searle is an assistant professor of Instructional Technology and Learning Sciences at Utah State University. Her work focuses on how participating in making activities (like e-textiles) can broaden teachers' and students' sense of what computing is and who can do it, with a particular focus on the development of culturally responsive computing pedagogies. Her work has appeared in publications such as Harvard Educational Review, Thinking Skills and Creativity and the International Journal of Multicultural Education. Bryan M. J. Brayboy is President's Professor of Indigenous education and justice in the School of Social Transformation at Arizona State University. At ASU, he is senior advisor to the president, director of the Center for Indian Education, associate director of the School of Social Transformation and co-editor of the Journal of American Indian Education. His research focuses on the role of race and diversity in higher education, the experiences of Indigenous students, staff and faculty in institutions of higher education. Yasmin B. Kafai is Lori and Michael Milken President's Distinguished Professor and Chair of the Teaching, Learning, and Leadership Division at the University of Pennsylvania. As a learning scientist and designer of online tools and communities that promote coding, crafting and creativity across grades K-16, her work empowers students to use computer programming to design games, sew electronic textiles and grow applications in biology with the goal of supporting creative expression, building social connections and broadening participation in computing. Address for correspondence: Breanne K. Litts, Instructional Technology & Learning Sciences, Utah State University, 2830 Old Main Hill, Logan UT 84322, USA. Email: breanne.litts@usu.edu*

### Abstract

Given the increased need for broadening participation in computing, there must be a focus not just on providing culturally relevant content but also on building accessible and inclusive computational tools. Most efforts to design culturally responsive computational tools redesign surface features, often through making nominal changes to add cultural meaning, yet the deeper structural design remains largely intact. We take a critical perspective towards novice programming environments to elucidate how the underlying structure privileges particular epistemologies and cultures. In this paper, we examine how the cultural practice of storytelling is supported and/or inhibited within novice programming tools. We draw upon the experiences of 38 Native American youth, who worked in teams to create place-based, interactive stories and games for their community. Findings offer insights to the embedded cultural biases that exist in the structures of computational tools. We discuss insights for how to address cultural biases and promote deeper integration of cultural practices in future designs of culturally responsive computational tools.

**Key words:** computational justice, computing, culture, design, equity, learning technologies, storytelling

### Practitioner Notes

What is already known about this topic?

- Culturally responsive computing connects computing content heritage and vernacular cultural practices.
- “Black boxing,” or lack of transparency in how it works, in computational tools makes it difficult for novices to enter computing cultures.
- Design tools are embedded with particular ways of being, knowing, valuing and doing.

What this paper adds?

- Thirty-eight novice learners’ computational designs were shaped by the ways in which a computational tool privileged particular knowledge systems.
- Storytelling, as a critical cultural practice, especially in Indigenous cultures, is heavily constrained by the design structure of computational tools.
- Computational tools are cultural artifacts with deeply embedded epistemological, ontological and axiological biases, which directly frame what learners can do with these tools.

Implications for practice

- Collaborative, community-based design processes could mitigate the cultural biases that persist in computational tools.
- Transparency in computation tools is critical to broadening participation in computing cultures.
- Culturally responsive design of computational tools at the structural level is required to build inclusive computing cultures.

### Introduction

The focus on youth of all cultures to become not just consumers but also producers of digital media has stressed the importance of introducing computing for all, especially now as CS education is introduced in K-12 schools across the globe (eg, Lindberg, Laine, & Haaranen, 2019). Similar to Manches and Plowman’s (2017) argument that the rate of change in computing education has outpaced the time it takes to develop corresponding researcher-informed pedagogy, we note a similar tension in the design of computational tools. Many pedagogical and computational tools have been developed that help novices to engage with computing, but the focus on computational concepts and practices alone, which, while critical, is not enough. It is equally important to focus on the social and cultural nature of computation to help *all* youth feel empowered to develop identities in a field that traditionally has been dominated by white men. Thus, there has been a shift from thinking about tools that support computational *thinking* to tools that support computational *participation* (Kafai & Burke, 2014) and *critical inquiry* (Kafai, Proctor, & Lui, 2020). From this perspective, we are equipping youth to not just be consumers of computational tools, but also producers and shapers who leverage the tools towards meaningful and critical societal participation.

To better understand which and how social and cultural factors hinder or support connections and participation, scholars have turned their attention to culturally responsive computing and justice-centered computing, which consider critical issues of access and equity with computational

tools. Culturally responsive computing focuses on connecting computing content with heritage and vernacular cultural practices or social justice issues that are familiar and/or meaningful to students (Eglash, Gilbert, & Foster, 2013; Scott, Sheridan, & Clark, 2015). This work also recognizes that youth need to become producers in digital arenas, which requires a shift not just in how we approach computing for minoritized youth, but also how we approach the tools themselves. The need to design for culturally responsive technology use, especially due to the tensions between western and Indigenous knowledge systems, is not new (McLoughlin, 1999).

Visual programming tools, which often use metaphors such as puzzles in block-based environments, have become a popular pathway for broadening participation in computing because of their low barrier to access (Resnick & Silverman, 2005). Current efforts to make visual programming tools culturally responsive include making changes to surface features such as content or element names, the context the tool is presented in, or the design of the computing activities themselves (eg, Eglash & Bennett, 2009; Eglash, Bennett, O'Donnell, Jennings, & Cintorino, 2006). These surface-level modifications result in a culturally responsive experience in the visual programming tool, but do not change the underlying structure of the tool itself, which is rooted in its own cultural perspectives and epistemologies. This hidden nature of the underlying functions and code creates what scholars call a "black box," which keeps the technology inaccessible to novices (Resnick, Berg, & Eisenberg, 2000). Hence, issues of cultural and implicit biases have come to the forefront in critical conversations about the structural design of technologies, especially in regard to how technologies mirror the biases of the humans that make or interact with them (Caliskan, Bryson, & Narayanan, 2017).

Taking seriously our need to make accessible and culturally responsive computational tools for novice programmers, we argue that deeper changes to the structure, perhaps in addition to surface-level modifications, produce more culturally responsive tools. One compelling example are computational platforms as culturally responsive tools embedded with the cultural practice of storytelling. Research on Indigenous storytelling argues that how we listen and tell stories are culturally based activities (Archibald, 2008; Brayboy, 2005; Brayboy, Gough, Leonard, Roehl, & Solyom, 2012). Here, we focus in particular on the relationship of storytelling-focused computational tools to the stories Indigenous youth wanted to tell. Specifically, our inquiry is guided by the research question: How do computational platforms support or inhibit Indigenous youth participants' in (re)creating their own cultural narratives? To honor tribal sovereignty, we include in this paper only data from one workshop with 38 Native American youth, which we are permitted to share with the public. Findings revealed that, while the computational platforms engaged novice programmers in computational thinking practices, the deep design structure of the platforms supported some narrative structures better than others. Building on existing work around the epistemological tensions of technology (Winter & Boudreau, 2018), we discuss insights for the design of culturally responsive computational tools, especially in regard to the embedded cultural biases that may exist in their deep design structures.

## **Background**

### *Computational participation and culturally responsive computing*

The recent K-12 Computer Science Framework (2016) highlights the need for creating, testing and refining computational artifacts and tools to promote inclusive computing cultures. Moreover, computational thinking (CT) has moved to the forefront of the conversation around integrating computing in classrooms (Wing, 2006). CT consists of cognitive and problem-solving skills that outline how humans can process information like a computer scientist (Grover & Pea, 2013). The National Research Council (2010, 2011) expanded notions of CT to include practices

beyond “just programming” and identified a need to develop pedagogical supports in novice computing environments. Thus, scholars are turning their attention to not only considering the design of computational cultures but also the design of computational tools.

Scholars have identified a need for more accessible interfaces with which to teach programming (Mendelsohn, Green, & Brna, 1990). Visual metaphors for text-based code have been widely adopted in education contexts. Logo Turtle (Papert, 1980), for example, was one of the first efforts to translate abstract text-based code to a visual and physical context. Since Logo, though, myriad platforms, approaches and models have emerged to make text-based code more accessible through visual metaphors. Resnick and Silverman (2005) encouraged in Scratch designing creative programming interfaces with a “low floor,” “high ceiling” and “wide walls” to support novices, experts and exploration, which prompted the design of visual development environments. The most common visual metaphor adopted in novice programming contexts leverages a block-based interface through which color-coded, puzzle-like blocks are compiled for programming. Block-based environments such as LogoBlocks (Begel, 1996), Alice (Cooper, Dann, & Pausch, 2000), Scratch (Resnick *et al.*, 2009), Snap! (Harvey *et al.*, 2014) and Blockly (Fraser, 2013) teach programming topics to youth across a range of contexts, including for mobile games with platforms like App Inventor (Morelli *et al.*, 2011) and TaleBlazer (Medlock-Walton, 2012).

While re-design of interfaces addresses an important dimension of making computing accessible, culturally responsive computing, considers issues of access and equity in computing cultures with computational tools. Culturally responsive computing focuses on connecting computing content with heritage and vernacular cultural practices and/or social-justice issues that are familiar to students (Eglash *et al.*, 2013; Scott *et al.*, 2015). Current research focused on making visual programming tools culturally responsive often modifies surface features such as element names, the context the tool is presented in, or the design of the computing activities themselves (eg, Eglash & Bennett, 2009; Eglash *et al.*, 2006). These surface changes result in a culturally responsive experience in the visual programming tool, which is a valuable and notable step in the right direction. McLoughlin (1999) refers to this level of design as the “micro cultural level,” and we argue that as technological innovations have continued to occupy much of our society, there is an urgent need to consider the design of technology at an even deeper level. Thus, we seek to push this work to consider the structure of the tool itself, which is embedded with its own cultural perspectives. Issues of cultural and implicit biases have come to the forefront in critical conversations about the structural design of technologies, especially in regard to how technologies mirror the biases of the humans that make or interact with them (Caliskan *et al.*, 2017). We bring these issues to conversations around culturally responsive computing by taking a critical perspective towards the design of computational platforms to illustrate how the cultural biases embedded in these tools support and/or inhibit cultural practices.

### *Technologies as epistemologies*

We argue that the epistemological foundations in the design of the tools support particular ways of being, knowing, valuing and doing (Brayboy & Maughan, 2009). In the context of programming languages, epistemological tensions are embedded in the design of the tools themselves. Most often these tools are designed with a Western orientation to thinking and programming in a linear, top-down fashion rather than allowing in a bottom-up tinkering approach (Turkle & Papert, 1990). Here, we emphasize the ways in which Indigenous Knowledge Systems can highlight the limitations of Eurocentric theory and provide a new perspective on our taken-for-granted notions of computational tool design. As Battiste (2002) notes:

Indigenous scholars discovered that Indigenous knowledge is far more than the binary opposite of western knowledge. As a concept, Indigenous knowledge benchmarks the limitations of Eurocentric theory—its methodology, evidence and conclusions—reconceptualizes the resilience and self-reliance of Indigenous peoples, and underscores the importance of their own philosophies, heritages and educational processes. Indigenous knowledge fills the ethical and knowledge gaps in Eurocentric education, research and scholarship. By animating the voices and experiences of the cognitive “other” and integrating them into educational processes, it creates a new, balanced centre and a fresh vantage point from which to analyze Eurocentric education and its pedagogies (p. 5).

Thus, we consider the design of computational tools guided by Indigenous Knowledge Systems, which are lived, adaptable and embodied in the lives of individuals and communities. We bring these issues to conversations around culturally responsive computing by critically examining how the cultural biases embedded in computational tools can support and/or inhibit particular ways of telling stories, which are rooted in particular ontologies, axiologies and epistemologies associated with different knowledge systems and cultures. By making this move, we join Richards and Dungham (2019) assertion to take an ethical and socially responsible stance in the design of learning technologies.

In this paper, we focus on how the cultural epistemologies of storytelling, a key element of Indigenous Knowledge Systems, are supported or constrained in computational production tools. Specifically, we are concerned with computational tools designed to support various forms of storytelling and its limitations when we consider culturally patterned narratives (Archibald, 2008; Brayboy, 2005; Brayboy *et al.*, 2012). Indigenous stories are often centered around place rather than characters (Basso, 1996) and it is incumbent upon the listener to understand the story rather than on the narrator to make the story clear (Archibald, 2008). Moreover, narratives take up various forms and structures from linear to branching to hub-and-spoke (Djwa, 2011; Schreiber, 2009). Taken together, place-based storytelling provides a unique context that illuminates critical epistemological tensions in this computational tool. To make culturally responsive computational tools accessible for novice programmers from Indigenous communities, we argue that deeper changes to the structure, in addition to surface-level modifications, can produce more culturally responsive tools.

## **Methods**

### *Partnership and positionality*

The nature of our partnership and our (the researcher’s) positionality within that partnership is significant. While our team is composed of both Indigenous and nonIndigenous researchers with a history of working with Indigenous communities, all were outsiders to the community we partnered with (Brayboy & Dehyle, 2000). Members of our team had worked with this specific American Indian community (10,000 enrolled members) in the Southwest part of the United States for 5 years prior to the workshop we present here. We had previously worked with the community on policy issues and in research and outreach capacities in schools and in the community more broadly, resulting in established relationships with tribal government officials, education department staff and culture department staff. Yet, in spite of these relationships, it is important to remember that we remained outsiders with differing epistemological perspectives.

We collaboratively designed and implemented (DiSalvo, Yip, Bonsignore, & DiSalvo, 2017) a summer workshop in partnership with various stakeholders. Stakeholders included the community’s education, cultural resources and public relations departments, as well as staff at the local community college where the workshop took place. The co-design process occurred over several months of regular meetings during which time the various community stakeholders identified

storytelling as a cultural practice they wanted to integrate into the workshop. Ultimately, the goal was that the youth's digital products would become something that the community's public relations department could use to share information with visitors to the community.

### Workshop

Forty-seven Native American youth participated in the summer workshop and 38 (23 girls, 15 boys, ages 12–14 years old) fully consented to participate in research. Participants' prior experience with game design and programming varied somewhat, but most did not have any prior experience with either domain. Participation in the summer workshop did not require any prior experience in either. The 7-day workshop was integrated into their summer camp with each day consisting of 1–2 sessions of 1–2 hours each. On Day 1, we introduced the scope of the project and youth were randomly divided into 15 groups of 2–4 students each. On Day 2, youth were divided into three larger groups (five groups per tour) and went on three different community tours (one with a Community sculpture artist and two with economic development tours highlighting tribally owned sites) to collect primary source data for their designed experiences. On Day 3, youth completed a storyboarding process using paper prototyping materials to build an analog version of their game. Days 4–7 youth used their storyboards as they built their stories in the computational platform (see Figure 1). Two members of the research team worked with two teachers and two teacher aides to facilitate the workshop, and six additional members of the research team rotated data collection responsibilities throughout the workshop.

### Computational tool

We chose Augmented Reality and Interactive Storytelling (ARIS), a novice programming platform to support location-based stories with augmented reality technologies, for the workshop (Holden, Gagnon, Litts, & Smith, 2014). ARIS is made up of a web-based editor available on any browser and an app-based client available on iOS. The unique location-based affordances of ARIS via an iPad or iPhone made it a suitable computational platform for the place-based stories our community partners envisioned that youth would create. With ARIS, youth can create a number of different types of location-based experiences, such as digital tours of specific places, field data collection, or Pokémon Go-like games. ARIS was designed to be an accessible platform for novice programmers and has been used across a range of settings and demographics and for a variety of purposes (eg, Dikkers, Martin, & Coulter, 2012; Holden, Dikkers, Martin & Litts, 2015).

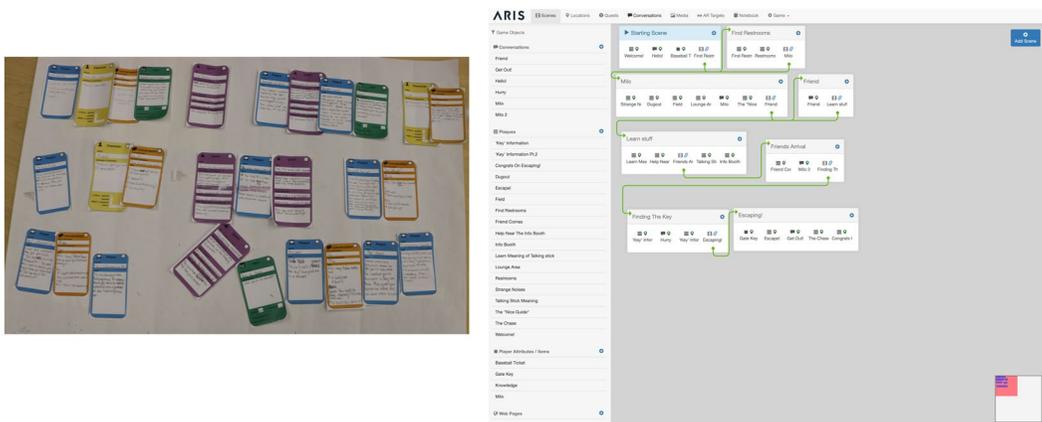


Figure 1: Example of a storyboard (left) and the final project in the editor (right) [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

Narrative research highlights that story elements also require elements of connectedness to construct a cohesive story (Bruner, 1991), so ARIS uses computational concepts like conditionals and functions to act as the connective tissue of the stories created with the tool. For example, in contrast to generic block-based metaphor that uses color-coded puzzle pieces, ARIS translates traditional computer science principles such as variables, conditional logic and flow control through narrative metaphors of “characters,” “events” and “scene transitions.” Furthermore, ARIS also allows youth to program with HTML or JavaScript as they become more comfortable with coding. We have explored elsewhere how this computational design process engaged youth with community (Litts, Lewis, & Mortensen, 2020; Searle, Casort, Litts, & Benson, 2017), but in this piece we address the cultural responsiveness of the computational tool itself, especially in regard to how it supports and/or inhibits the use of storytelling as a cultural practice.

### *Data collection & analysis*

We collected a range of qualitative data aimed to help us better understand the youths’ making process in relation to the ARIS tool. Data sources included field notes, audio recordings, interview transcripts, photographs of participants’ making processes, design artifacts from each group including their storyboards and screenshots of their in-process games, and their completed ARIS games. To analyze these data, we used a bidirectional artifact analysis technique presented by Halverson and Magnifico (2013). Bidirectional artifact analysis is a process-centered analytic framework that connects the participant’s design artifacts, interviews and observations to recreate their design process through backward and forward lenses of time. Unlike descriptive analyses that typically move forward, we find it important to move bidirectionally from the final product to their initial idea and from their initial idea to their final product. In our analysis, we not only examined how early design artifacts laid a foundation for later progression, but also how later design work reflects initial ideas. Specifically, we trace across data types (eg, design work, audio recordings, etc) forward from initial brainstorming and storyboarding to the completed games as well as backwards from the completed games to the initial conception. In this process, we note key challenges, decision points and changes along with their rationale. By bidirectionally connecting learners’ work and what the learners say about their work, we are able to get a more holistic understanding of when, how and why learners made key design choices. Practically, this means we constructed cases primarily for *intrinsic* purposes to develop a deep understanding of groups’ design processes and secondarily for instrumental purposes to understand how these design processes inform the design of novice computational platforms (Stake, 2008). Cases were bound by the group-level design process and were constructed with visual representations that connected all data across the collaborative design process for 14 groups (a 15th group did not agree to be part of the research study). In the findings, we focus on three cases, because they depict unique narrative structures that revealed interesting interactions with the ARIS tool. We use pseudonyms for all participants.

### **Findings**

Across projects youth developed a range of games, some connected to their cultural heritage and others connected to current popular culture. Thus, while all games were connected to the physical locations of the youths’ respective tours, they varied significantly in genre and, thus, tended to vary in narrative structure. For example, games ranged from finding a lost dog at a local golf course to a Zombie boarding school, to a scavenger hunt about tribal culture. Here, we present three illustrative cases, which we selected to show the range of products and the three forms of underlying narrative structures youth designed across all groups. Though we observed these three narrative structures, most groups adopted the branching narrative structure supported by ARIS. Within these cases, we highlight the tensions between the youths’ idea and the affordances

and constraints of ARIS as a computational tool, especially in regard to how the tool supports and/or inhibits storytelling as a cultural practice.

### *Branching narrative*

The narrative structure that is most explicitly supported by the logic in ARIS is the branching narrative structure, which follows tree-logic (Schreiber, 2009). Group 11 consisted of two girls, Selma, Tess and another individual who was not part of the research study. The girls described their collaboration as Tess taking the lead on building in the ARIS editor while Selma and the other individual focused on testing on the iPad and storyboarding. They participated in the same economic development community tour as Group 12 (described below). They took a playful approach and developed a story during which the player is trapped in the park and has to figure out how to get out. Selma summarized their game narrative:

Basically, it's supposed to be going to go check out the fields, and then the actual tour guide, Aaron, he gives you a ticket, and then later on in the game you find out that—well, you find the dog, and there's a power out when you find the dog. You use your flashlight on your phone. Then, he tells you that Aaron's a bad guy and that you can't return the ticket, and it keeps you inside of the whole place until you find three knowledge, you earn three knowledge points for anything that you learn. Then, he just tries to get you out cuz if he doesn't get you out, then you will turn into a dog like him. That's why he's like that and he's stuck in there. He tries to make you escape and gives you hints to go around places to find more knowledge (Interview, June 15, 2017).

To enact this idea, they employed the branching narrative structure embedded in the ARIS platform through the conversation element (see Figure 2). As Tess described, “you have the choice to either just leave the park then, before he gives it to you, or just continue on what he says. You have that choice... then, you have the choice to either slowly back away, run away, or approach Milo [the dog]” (Interview, June 15, 2017). Group 11 integrated this branching structure throughout their game consistently using the tree-logic in the ARIS conversation element to give the player choices (see Figure 2).

This branching in the conversations combined with the logic gave this game a linear flow where the player was guided through the story, or from one element to the next. Though Group 11 encountered typical challenges one might expect in designing their first computational artifact, their idea persisted from their storyboard to their final game. When transitioning from the narrative on their storyboard to the initial development of their game, they effectively translated their idea into the editor since it aligned with the logic embedded in the platform.

### *Sandbox narrative*

Another narrative structure that we observed is the sandbox narrative structure (Picucci, 2014), which often resulted due to a lack of transparency in ARIS. A sandbox narrative structure is one through which the player freely engages with any/all elements of the game. With Group 12, which consisted of two girls, Gina and Abigail, participants' design process was heavily affected by the assumed knowledge of computer science embedded throughout the platform. We demonstrate this below with Group 12's final project (see Figure 3). This group was unable to identify the needed elements and logic to fully realize their idea leading to an unintentional sandbox story structure.

Gina and Abigail, who described their collaboration as Gina primarily building the game in the ARIS editor and Abigail primarily testing the game on the iPad (Interviews, June 15, 2017). Gina and Abigail went on an economic development community tour that included two tribally owned sites: the local baseball fields and a skydiving facility. Using this field trip as inspiration,

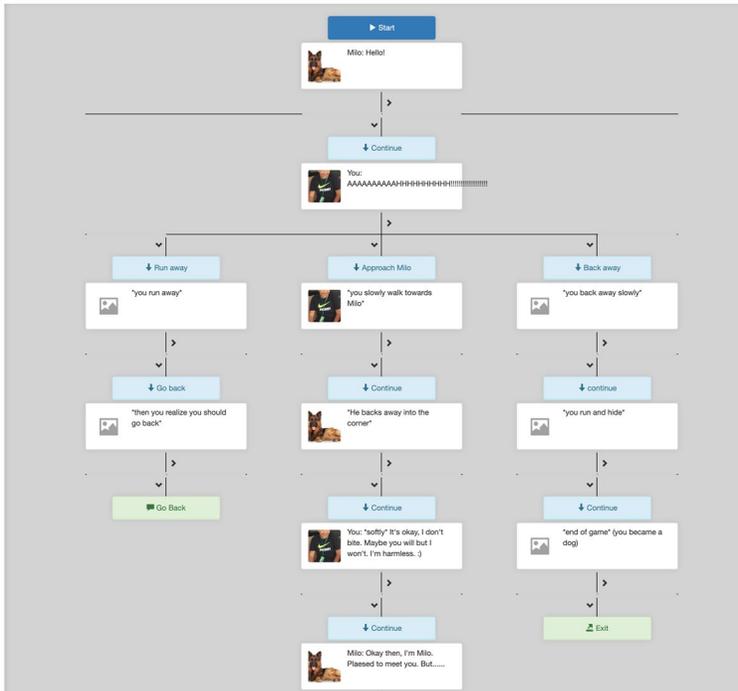


Figure 2: Example of how Group 11 used existing branching logic in the ARIS editor [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

Gina explained their game idea saying, “It’s a game where you have to —I don’t know. It’s kind of like a map, and you have to look for... You have to look for different types of things on the map and try and solve the problem” (Interview, June 15, 2017). Abigail elaborated on their game idea: “Well, there’s people at the gate. They went to the game, and then they dared each one of ‘em to stay a night. Then one of them—someone comes after each one of ‘em and—yeah. Like they’re in the field for overnight” (Interview, June 15, 2017). This idea persisted through the group’s storyboard to their final product, however, they struggled to effectively enact their idea in the ARIS platform.

Cohesion is a crucial element when developing a narrative and in computational environments, elements such as conditional logic and flow control are tools for connecting ideas and variables to achieve cohesion. In the case of Group 12, they used arrows to show this connection on their storyboard, however, they struggled to translate these connections to the ARIS editor. Gina explains that their biggest challenge was figuring out “how to put everything together” in the ARIS editor (Interview, June 15, 2017). Abigail elaborated on this tension: “I just don’t know how to work it, put in certain things in certain places, like plaques and the characters and conversation and the quest” (Interview, June 15, 2017). Translating an ARIS experience from storyboarding into the editor has shown to be a challenge in other contexts (Gagnon, Vang, & Litts, 2015; Litts, Martin, & Gagnon, 2013), because it requires a certain level of familiarity and expertise with not only the ARIS platform but also computer science principles.

In their final game, though, all ARIS elements are available for the player to explore from the start, which is an exploratory, sandbox experience for the player through which progression is

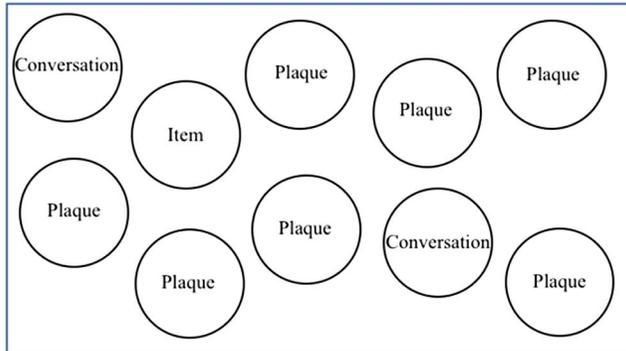


Figure 3: A visual representation of Group 12's sandbox narrative structure [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

dependent on the player freely exploring the different elements. Abigail and Gina's sandbox structure is easily supported in the ARIS editor, but it was a result of the lack of transparency of the logic and sequencing tools they needed to connect the ARIS elements rather than an intentional decision to create a sandbox narrative. Their final product included their narrative without cohesion or a way to guide the player through the game, which resulted in a sandbox narrative structure they did not initially intend.

#### *Hub-and-spoke narrative*

The final narrative structure youth employed was the hub-and-spoke structure, which is a non-linear structure anchored around a central hub rather than particular order (Djwa, 2011). ARIS does not easily or explicitly support the hub-and-spoke narrative structure. Curiously, the one group who employed this structure also made the game most tightly connected to their cultural heritage. Consider the case of Group 2, which consisted of three girls, Julia, Carey and Melissa. They described their collaboration as Julia taking the lead in building their story in the ARIS editor with help from Melissa and Carey and Melissa doing most of the testing on the iPad (Interviews, June 15, 2017). They participated in the tour with the Community sculpture artist, who shared both the stories behind the sculptures and the materials he used to create them. Julia, Carey and Melissa used this tour to build their game idea, which Carey summarized:

There's four characters. We picked four sculptures out of all the sculptures that we viewed. Then we choose [sic] three materials that were used making the sculpture. The player's going to have to find the materials that were used to create the sculpture. Within that, they can find out where it's located, as well (Interview, June 13, 2017).

Their idea resulted in the development of a hub-and-spoke narrative structure (see Figure 4) through which the artist would act as the hub and the collection of the sculpture materials would function as the spokes, and the player would circle back to the hub over and over. Melissa explained further that "after [the player] finds the materials they go back to the artist and the artist creates it for them and then show[s] them a picture and tells them the story about his artwork" (Interview, June 15, 2017). So, the player starts by speaking with the artists and choosing a character (the hub), which determined what sculpture materials they would be searching for (the spokes). Once the materials were collected, the player would go back to the artist to learn about the sculpture and select a new character, or new spoke.

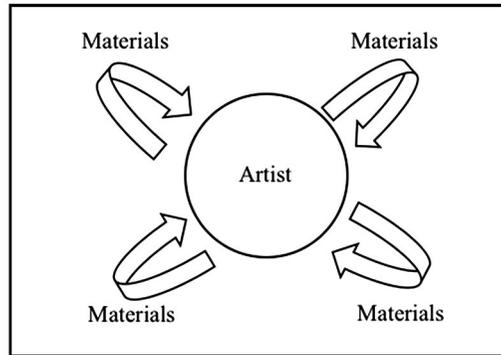


Figure 4: A visual representation of Group 2's hub-and-spoke narrative structure

Group 2 was able to clearly lay out their vision in their storyboard, or analog representation of their narrative, such that the ideas and how those ideas would be reconciled using ARIS were clearly defined. Carey elaborated that they wanted “to do something different, so we ended up doing the materials instead of just finding the sculpture on its own” (Interview, June 13, 2017). Interestingly, their desire to do “something different” revealed a deep tension with the ARIS platform.

Each member of this group shared that creating their game idea in ARIS was difficult. Carey summarized these difficulties: “to do one part and then go back and then redo it, and then we have to do it over again with all of them. We have to practice on one and see how that will come out, and then we used the table to test it. Then if anything went wrong, then we'd go back and redo it” (Interview, June 13, 2017). Her description refers to being able to dictate the flow of the game with the logic in ARIS, which proved to be a tension for these girls even though they identified the correct elements to use in their storyboard. This is represented in their final game as there was only one completed spoke that worked. Once completing the built out spoke, the player can get back to the hub of the game, but the remaining spokes are not playable. Furthermore, if this group were to build out each spoke, they would have had to spend time finding ways to “hack” the tool to fully realize their narrative ideas, since ARIS is not currently built to support a narrative that returns to a hub. Thus, the structure of the tool did not support the story structure of this group's narrative.

## Discussion

In this paper, we contribute to a better understanding of culturally responsive computing by considering the situatedness of the deep structural design features of computational tools. We examined the affordances and constraints of using a novice computing platform for community storytelling in a summer workshop with Indigenous youth. While all youth groups were able to design location-based stories incorporating community landmarks and art, our analysis also revealed that the design of the computational tool itself affords some narrative structures while constraining others. We examined these biases within the context of ARIS, but insights gained most likely also apply to many other computational tools (Litts, Searle, Hamilton, Yazzie, & Mannie, 2020). Thus, lessons learned inform how we design technologies for education contexts more broadly. In the following sections, we discuss two critical aspects of embedded bias that designers of future computational tools need to take into account to be more culturally responsive.

### *Hacking the narrative structure*

The reality of most computational tools is that they can be hacked to achieve any particular purpose. In other words, the logic in ARIS can be hacked to create any of the narrative structures the youth in our workshop attempted; they are indeed *possible* in ARIS. The issue, though, is how certain types of stories are much easier to create than others, which reflects an underlying cultural bias in the tool towards linear narratives. This is particularly problematic in tools created for use by novice and non-programmers with a goal of broadening participation in computing. It is not enough just to broaden who participates in computing; we must also broaden the epistemologies and cultural practices that are valued within computing spaces. As such, we must expand *who* designs these tools in the first place.

We might expect that individuals new to programming are unlikely to hack the tool on their first try. For example, to execute Group 2's hub-and-spoke narrative, they needed a sophisticated understanding of the nuances of the computational platform as well as computer science principles, which they did not possess. At the other end of the spectrum, Group 12 effectively designed an experience based on branching narratives with much more ease, since this tree-logic is embedded in the platform. Both groups had the same levels of knowledge of the tool and novice-level expertise in computing, but due to the way in which the tool privileges certain epistemologies, Group 2 was fundamentally disadvantaged in their computational design process. This deeper structure issue with novice computational tools reinforces persistent inequities in computational cultures by demanding expertise to enable expression.

Our point here is not that computational tools are inherently "bad," but that we need to recognize the cultural biases embedded within these computational tools and how this affords or constrains certain types of activities. The youths' task was to tell a cohesive story one could follow, but that story needn't be linear or branching in its timeline or structure. Even though computer science logic follows Boolean or if-then logic, many great stories jump around from the present, to the past and to the future. One might even argue that real life happens linearly, but that our engagement and recollection of it is not. We recognize this disconnect between time as a scale of linearity (or branching) and life as disjointed (or hub-and-spoke) as an opportunity to account for this variation and design more culturally responsive tools to better support diverse approaches and interpretations.

### *Transparency in computing*

Algorithms and technologies that continue to shape our experience in the world remain inaccessible and, in some cases, are intentionally opaque to disinvite participation from minoritized groups. Many electronic devices and aspects of coding are "black boxed," meaning that how they work is hidden from view. Designers of learning technology challenge us to design beyond these "black box" instruments by making tools that are at the right "level" of understanding, making key concepts visible as appropriate to learning goals (Resnick *et al.*, 2000). In this paper, we argue that designing computational tools for transparency is not only about designing for the right level of expertise but also in a way that makes space for multiple ways of knowing about the world. Hence, we argue that a critical pillar of cultivating diverse computing cultures is to design tools that easily support multiple epistemologies, ontologies and axiologies. One way to design such supports is to create transparent tools where the logic is easily visible to novice or non-programmers.

A key insight from our study highlights the practical impacts of lack of transparency. Unlike the branching structure Group 12 used to give the player choice in the story, which visually showed the path the players would take, much of the logic in ARIS has no visual representation. Thus,

the tool allowed no real outlet to Groups 2 or 11, thereby diminishing their process and product. Indigenous storytelling practices are just one context for cultural biases, but the issue of transparency in computational tools is crucial, especially when designing for cultural responsiveness. Designing for transparency rather than embedding or assuming knowledge of computer science principles or privileging certain types of cultural practices will afford more flexible computing environments and is a critical consideration in designing for inclusive computing cultures. The “black box” is not unique to novice computational tools, but permeates much of our technological and computational worlds (O’Neil, 2016; Richards & Dignum, 2019; Winter & Boudreau, 2018).

#### *Limitations and directions for future research*

Our study was heavily informed and guided by our partnership with a specific Indigenous community and in a particular context. The computational limitations we note in this paper may be specific to this community and through further research with other communities, who ascribe to different knowledge systems and ways of being, scholars and designers can better understand these nuances of the tool constraints. That said, the ways differences in knowledge systems and environments result in communities taking up tools differently is well-documented (eg, Bang *et al.*, 2013; Kawagley, 1995; Litts, Searle, *et al.*, 2020), so we believe the computational limitations outlined in this paper may resonate with other communities who prefer nonlinear narrative storytelling. Additionally, in this paper, we share findings from a single iteration of the workshop, which provide critical in-depth insights, yet there remain opportunities to gather observations across multiple workshops. Finally, the focus of our analysis in this paper is on the tool itself, however, it is equally important to also consider the culturally responsiveness of the design of the workshop itself, which we are currently exploring in other work.

#### **Conclusion**

The design of learning technologies is a core pillar of education research. Our study highlights the cultural biases embedded in not only the structure of what we design but also the ways in which we design. A critical takeaway of our study is that computational tools need to be in a process of continual iteration, particularly through a co-design process with diverse communities to identify cultural biases particularly in structural design features. Thus, as we look to expand access to and broaden participation in computer science moving forward, it is critical that we are designing computational tools that are culturally responsive across communities and contexts. This also includes gaining insights from youth as creators to know how they might hack the platform to create a more flexible and transparent structure in which they can better achieve their design ideas and goals.

#### **Acknowledgements**

This work was supported by a collaborative research grant (#1623453/1623404) from the National Science Foundation to Bryan Brayboy, Kristin Searle, Breanne Litts and Yasmin Kafai. Opinions, findings and conclusions expressed are those of the authors and do not necessarily reflect the views of our respective institutions or NSF.

#### **Statements on open data, ethics and conflict of interest**

For the protection of the Indigenous youth in the study and to honor tribal sovereignty over the data, we will not make data available to other researchers outside of published findings from our

work. The Institutional Review Board at Arizona State University approved this study. Informed parental consent and youth assent were obtained from all participants. The authors have no conflict of interest.

## References

- Archibald, J. A. (2008). *Indigenous storywork: Educating the heart, mind, body, and spirit*. Vancouver, BC: UBC Press.
- Bang, M., Marin, A., Faber, L., & Suzukovich III, E. S. (2013). Repatriating indigenous technologies in an urban Indian community. *Urban Education, 48*(5), 705–733.
- Basso, K. H. (1996). *Wisdom sits in places: Landscape and language among the Western Apache*. Albuquerque, NM: University of New Mexico Press.
- Battiste, M. (2002). *Indigenous knowledge and pedagogy in First Nations education: A literature review with recommendations*. Ottawa, ON: National Working Group on Education.
- Begel, A. (1996). LogoBlocks: A graphical programming language for interacting with the world. In: *Electrical Engineering and Computer Science Department*. Boston, MA: MIT.
- Brayboy, B. M., & Deyhle, D. (2000). Insider-outsider: researchers in American Indian communities. *Theory into Practice, 39*(3), 163–169.
- Brayboy, B. M. K. J. (2005). Toward a tribal critical race theory in education. *The Urban Review, 37*(5), 425–446.
- Brayboy, B. M. K. J., Gough, H. R., Leonard, B., Roehl, R. F., & Solyom, J. A. (2012). Reclaiming scholarship: Critical indigenous research methodologies. In S. D. Lapan, M. L. T. Quartoli, & F. J. Riemer (Eds.), *Qualitative inquiry* (pp. 423–450). San Francisco, CA: Jossey-Bass.
- Brayboy, B. M. J., & Maughan, E. (2009). Indigenous knowledges and the story of the bean. *Harvard Educational Review, 79*(1), 1–21.
- Bruner, J. (1991). The narrative construction of reality. *Critical Inquiry, 18*(1), 1–21.
- Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science, 356*(6334), 183–186.
- Cooper, S., Dann, W., & Pausch, R. (2000). Alice: A 3-D tool for introductory programming concepts. *Journal of Computing Sciences in Colleges, 15*(5), 107–116.
- Dijkers, S., Martin, J., & Coulter, B. (2012). *Mobile media learning: Amazing uses of mobile devices for learning*. Pittsburgh, PA: ETC Press. <http://repository.cmu.edu/etcpres/62>.
- DiSalvo, B., Yip, J., Bonsignore, E., & DiSalvo, C. (Eds.) (2017). *Participatory design for learning: perspectives from practice and research*. New York, NY: Routledge.
- Djwa, P. (2011). *White Paper: Interactive narrative - An emerging form*. Retrieved from <http://www.agentic.ca/blog/white-paper-interactive-narrative-emerging-form>.
- Eglash, R., & Bennett, A. (2009). Teaching with hidden capital: agency in children's computational explorations of cornrow hairstyles. *Children Youth and Environments, 19*(1), 58–73.
- Eglash, R., Bennett, A., O'Donnell, C., Jennings, S., & Cintorino, M. (2006). Culturally situated design tools: ethnocomputing from field site to classroom. *American Anthropologist, 108*(2), 347–362. Retrieved from <http://www.jstor.org/stable/3804796>.
- Eglash, R., Gilbert, J. E., & Foster, E. (2013). Toward culturally responsive computing education. *Communications of the ACM, 56*(7), 33–36.
- Fraser, N. (2013). *Blockly*. Google. <https://code.google.com/p/blockly>.
- Gagnon, D., Vang, R., & Litts, B. K. (2015). *Proceedings of the eleventh annual Games+Learning+Society Conference* (pp. 282–288). Pittsburgh, PA: ETC Press.
- Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43.
- Halverson, E., & Magnifico, A. (2013). Bidirectional artifact analysis: A method for analyzing digitally mediated creative processes. In R. Luckin, S. Puntambekar, P. Goodyear, B. Grabowski, J. Underwood, & N. Winters (Eds.), *Handbook of design in educational technology* (pp. 406–415). New York, NY: Routledge.

- Harvey, B., Garcia, D. D., Barnes, T., Titterton, N., Miller, O., Armendariz, D., ... Paley, J. (2014, March). Snap! (build your own blocks). In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (pp. 749). New York, NY: ACM. <https://dl.acm.org/doi/10.1145/2445196.2445507>
- Holden, C., Dikkers, S., Martin, J., & Litts, B. (2015). *Mobile media learning: Innovation and inspiration*. Pittsburgh, PA: Carnegie Mellon University, ETC Press. <http://repository.cmu.edu/etcpres/31>.
- Holden, C. L., Gagnon, D. J., Litts, B. K., & Smith, G. (2014). ARIS: An open-source platform for widespread mobile augmented reality experimentation. In *Technology platform innovations and forthcoming trends in ubiquitous learning* (pp. 19–34). IGI Global.
- K–12 Computer Science Framework. (2016). Retrieved from <http://www.k12cs.org>.
- Kafai, Y. B., & Burke, Q. (2014). *Connected code: Why children need to learn programming*. Cambridge, MA: MIT Press.
- Kafai, Y. B., Proctor, C., & Lui, D. (2020). From theory bias to theory dialogue Embracing cognitive, situated and critical framings of computational thinking in K-12 CS education. *ACM Inroads*, 11(1), 44–53. <https://doi.org/10.1145/3381887>
- Kawagley, O. (1995). *A Yupiaq worldview: A pathway to ecology and spirit*. Prospect Heights, IL: Waveland Press.
- Lindberg, R. S., Laine, T. H., & Haaranen, L. (2019). Gamifying programming education in K-12: A review of programming curricula in seven countries and programming games. *British Journal of Educational Technology*, 50(4), 1979–1995.
- Litts, B. K., Lewis, W. E., & Mortensen, C. K. (2020). Engaging youth in computational thinking practices through designing place-based mobile games about local issues. *Interactive Learning Environments*, 28(3), 302–315.
- Litts, B. K., Martin, J., & Gagnon, D. J. (2013). *Learning through design with ARIS iteration 1: Digital literacies course* [white paper]. Madison, WI: Mobile Learning Incubator.
- Litts, B. K., Searle, K. A., Hamilton, M., Yazzie, K., & Mannie, C. (2020). Diversifying computational tools: How technology affords and constrains indigenous youths' agency. In *Proceedings of the Fourteenth International Conference of the Learning Sciences*.
- Manches, A., & Plowman, L. (2017). Computing education in children's early years: A call for debate. *British Journal of Educational Technology*, 48(1), 191–201.
- McLoughlin, C. (1999). Culturally responsive technology use: developing an on-line community of learners. *British Journal of Educational Technology*, 30(3), 231–243.
- Medlock-Walton, M. P. (2012). *TaleBlazer: A platform for creating multiplayer location based games*. Boston, MA: (Doctoral dissertation). Massachusetts Institute of Technology.
- Mendelsohn, P., Green, T. R. G., & Brna, P. (1990). Programming languages in education: The search for an easy start. *Psychology of Programming*, 175–200.
- Morelli, R., De Lanerolle, T., Lake, P., Limardo, N., Tamotsu, E., & Uche, C. (2011). Can android app inventor bring computational thinking to k–12. In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education (SIGCSE'11)* (pp. 1–6).
- National Research Council. (2010). *Report of a workshop on the scope and nature of computational thinking*. Washington, DC: The National Academies Press. <https://doi.org/10.17226/12840>.
- National Research Council. (2011). *Report of a workshop on the pedagogical aspects of computational thinking*. Washington, DC: The National Academies Press. <https://doi.org/10.17226/13170>.
- O'Neil, C. (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. New York, NY: Broadway Books.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books Inc.
- Picucci, M. A. (2014). *When video games tell stories: A Model of video game narrative architectures*. Retrieved from <http://revistacaracteres.net/revista/vol3n2noviembre2014/model-video-game-narrative-architectures/>.
- Resnick, M., Berg, R., & Eisenberg, M. (2000). Beyond black boxes: Bringing transparency and aesthetics back to scientific investigation. *The Journal of the Learning Sciences*, 9(1), 7–30.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60–67.

- Resnick, M., & Silverman, B. (2005, June). Some reflections on designing construction kits for kids. In *Proceedings of the 2005 conference on Interaction design and children* (pp. 117–122).
- Richards, D., & Dignum, V. (2019). Supporting and challenging learners through pedagogical agents: Addressing ethical issues through designing for values. *British Journal of Educational Technology*, 50(6), 2885–2901.
- Schreiber, I. (2009). *Game design concepts: An experiment in game design concepts [web blog]*. Retrieved from: <https://gamedesignconcepts.wordpress.com/2009/07/30/level-10-nonlinear-storytelling/>.
- Scott, K. A., Sheridan, K. M., & Clark, K. (2015). Culturally responsive computing: A theory revisited. *Learning, Media and Technology*, 40(4), 412–436.
- Searle, K. A., Casort, T., Litts, B. K., & Benson, S. (2017). Connecting space and narrative in culturally responsive making in ARIS with indigenous youth. In *Proceedings of 2017 FabLearn Conference on Creativity and Fabrication in Education*. ACM.
- Stake, R. E. (2008). Qualitative case studies. In N. K. Denzin, & Y. S. Lincoln (Eds.), *Strategies of qualitative inquiry*, 3rd ed. (pp. 119–150). Thousand Oaks, CA: Sage.
- Turkle, S., & Papert, S. (1990). Epistemological pluralism: Styles and voices within the computer culture. *Signs: Journal of Women in Culture and Society*, 16(1), 128–157.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
- Winter, J., & Boudreau, J. (2018). Supporting self-determined indigenous innovations: Rethinking the digital divide in Canada. *Technology Innovation Management Review*, 8(2), 38–48.