Confusion Over Models: Exploring Discourse in a STEM Professional Development

Todd Reynolds¹, Andrea C. Burrows¹, and Mike Borowczak¹

Abstract
For 2 weeks in the summer of 2018, K-12 STEM teachers (n=40) attended a professional development (PD) that included four sessions focused on computer science modeling with follow-up academic year sessions; however, overall, the teachers did not meet expectations about what modeling means or how to utilize it. To examine why, the authors looked at the teachers’ discourse. Using three theories to connect to the practice (terministic screens, schema theory, and concepts of capital) the authors analyzed the surveys, interviews, and email reflections to explore participants’ concept of models and the potential difficulties of implementing computer modeling in their classrooms. Five general concepts of models are presented. Findings show that the term modeling was interpreted differently by the PD’s faculty team and participants. Further, the authors found that the majority of presenters held differing theories of models than the participants. Participant concepts of models did improve slightly. Implications are provided showcasing articulated keys for delivering PD that assists in eliminating discursive and theoretical issues. Included are considerations for STEM teacher educators, PD providers, and K-12 teachers.

Keywords
teacher professional development, modeling, computer science, STEM, data, schema theory

At the end of 2018, the second author reflected on the summer’s professional development (PD) for K-12 science, technology, engineering, and/or mathematics (STEM) teachers. One goal of the PD was to assist K-12 teachers use computer science modeling in their classrooms. Combined with a state-wide initiative mandating computer science in K-12 classrooms by the 2022 to 2023 school year (https://edu.wyoming.gov/wp-content/uploads/2022/04/2022-WDE-Statutory-Changes-in-Computer-Science-Education.pdf), she believed these PD sessions would be well received, and that teachers would be able to integrate computer science in their classrooms once they finished the experience. However, she reflected:

I didn’t have many teachers reach out for modeling and/or computer science support this year, and that was worrisome to me. I emailed the group several times, and I expected the teachers to ask questions or bounce ideas off me (or someone from Robotics, Applied Mathematics, Physics, & Engineering Design (RAMPED) project team, but instead, it was almost silent on our end for help extensions even after we reached out and offered to even go into the classrooms.

The authors reflected together and asked why the PD results were different than expected in relation to modeling. The purpose of this article is to examine possible reasons for the disconnect between the faculty (part of the PD project team) and the teacher participants in regard to computer science, modeling, and eventual classroom implementation based on the discourse that was used during the PD. The research question guiding the PD inquiry:

- How did the “modeling” language used during the professional development impact the teacher participants and faculty (who promoted computer science modeling practices) when they returned to instructing?

After applying Burke’s (1966) terministic screens, schema theory (Tracey & Morrow, 2012), and Bourdieu’s (1986) concept of capital, and analyzing transcripts from interviews, pre/post surveys, and emails, the authors discovered that the term modeling created confusion for the teachers that may

¹University of Wyoming, Laramie, USA

Corresponding Author:
Andrea C. Burrows, College of Education—Secondary Science, University of Wyoming, 1000 East University Avenue, Dept. 3374, Laramie, WY 82071, USA.
Email: Andrea.Burrows@ucf.edu
have led to difficulties with practice; additionally, teachers may have desired a different type of capital as a result of attending the 2-week PD with six follow-up sessions during the academic year, than the instructors anticipated. These discursive issues may have contributed to the differences between the expectations and the participants’ actual application in their classrooms, but they also provide some guidance for moving forward with teacher PD. This article highlights considerations for successful PDs, context of the PD that was presented with computer science modeling, the research study itself, and the lessons learned including five general concepts of “models” and their uses.

**Traits of Quality Professional Development**

In teacher PD, Garet et al., (2001) argued that there are two main models: a traditional model and a reform model. The traditional model is top-down and aims to improve perceived needs in the skills of a faculty; a reform model focuses on building up the faculty through collaborative and classroom-embedded practices. The traditional model tends to be criticized, and not considered to be overly effective, while the reform model, with active learning, may be more in line with how teachers learn new material (Garet et al., 2001). However, it is not this dichotomy that is important, but the methods used during the PD. Regardless of the approach, some factors for effective PD appear consistent. The PD experience should be focused on content and how the participants learn, design, and deliver in a collaborative and active format that fosters active creation of knowledge instead of merely receiving knowledge, and conducted throughout the year in ways that embed practice in the participants’ classrooms (Abrahams et al., 2014; Garet et al., 2001; McGee et al., 2013). Burrows, (2015) found that effective PD should include: “clear communication, hands-on activities, planned time for reflection and discussion, and intentional partnership building” (p. 35).

However, another study Abrahams et al. (2014) noted that self-reporting of the participant PD experiences could be positive, but that the positivity did not necessarily translate to improved practices. Wilkinson et al. (2017), similarly, found that even though some of the teachers’ practices changed after a year-long PD, teachers’ beliefs did not show a correlating change, leading them to wonder if the teachers would revert back to their previous methods after the PD was completed. Burrows et al., (2016) also found that enthusiasm ebbed and flowed during a PD, with high engagement at the beginning and end, with a dip in enthusiasm in the middle of the session; as a result, the RAMPED project team strived to explicitly articulate the PD processes.

The principles of effective PD, and potential discrepancies between perception and practice factored into this study. This project RAMPED PD was a mix of a traditional and a reform model. The participants traveled to a regional location to study for 2 weeks; however, the learning was collaborative and active. Participants actively engaged in computer science modeling activities throughout the 2-week session, and they were provided with the opportunity to engage with the faculty throughout the year and years to come. As the authors reflected over the last 2 years, although the participants self-reported to enjoy their PD experience, unfortunately, they did not seemingly enact significant changes to their classrooms. While this trend is not unique, it did leave the authors questioning whether the modeling language used created some of the modeling disconnect, since an enormous amount of planning went into creating the active PD spaces to ensure authentic engagement, meaningful learning, and partnership building (Burrows, 2015; Burrows et al., 2016; Loukes-Horsley et al., 2010; Shernoff et al., 2017; Srikoom et al., 2018).

**Computational Modeling**

For this project RAMPED PD, the term **model** immediately set the various faculty involved on separate paths: computer science data models (the intended use), teacher modeling practice (e.g., I do -we do -you do), teaching and framework models (e.g., learning, process, concept, logic, and change), harnessing role models (e.g., Bell et al., 2010; Faulkner & Latham, 2016; Windschitl, 2004), and creating projects as models. See Figure 1 for an overview of these five concepts of modeling. Computer science modeling (what the project PD focused on) is a concept that can involve computational thinking and often promotes transformation and visualization of data (Borowczak & Burrows, 2019; Burkins & Yaris, 2019; Lovett & Forbus, 2017; Psycharias, 2018). To that end, computer science is a combination of problem-solving and optimization for domain specific applications, while balancing the societal and ethical implications of implementing algorithms as software often executed on real-world hardware platforms (Borowczak & Burrows, 2019; Lovett & Forbus, 2017; Psycharias, 2018). Additionally, computational thinking is specifically targeted reasoning through complex problems (Borowczak & Burrows, 2019; Grover & Pea, 2018; NRC, 2010; Wing, 2008). Several authors argue that computer science and computational thinking are separate disciplines (Barr & Stephenson, 2011; Kotsopoulos et al., 2017). As the STEM integration experts, the PD team promoted computational thinking as a component of computer science modeling, which includes the skill set of creating a representation of an algorithm (making a model). The representation of the algorithm manifesting in the model can be either theoretical (e.g., climate change models) or practical (e.g., new building construction). Wing (2008) explained computational thinking as that it is an “approach to solving problems, designing systems and understanding human behavior that draws on concepts fundamental to computing” (p. 3717).

The concept is complex as it draws several connected ideas into direct and supportive contact with the other. It has...
been described it as “the application of high level of abstraction and an algorithmic approach to solve any kind of problem” (Weintrop et al., 2016, p. vi). For classroom purposes, it can be described “a definition of computational thinking for mathematics and science in the form of a taxonomy consisting of four main categories: data practices, modeling and simulation practices, computational problem solving practices, and systems thinking practices” (Weintrop et al., 2016, p. 127). Computer science modeling is an important concept in STEM as it showcases the ability to display evidence, but computer science and computational thinking have been largely absent in the K-12 as well as higher education sectors (Sentence et al., 2018). With all the quickly changing pieces of a larger computer science puzzle, integration is on the horizon (Johnson et al., 2020).

Others discuss embodied modeling, which “introduces the students to the relevant computational rules represented by the agent-based programming commands, [and] helps them debug their programs and deepens their understanding of the graphs in the simulation” (Sengupta et al., 2018, p. 17). In other words, embodied modeling engages the student through the active learning of working with the problems, experiencing the programming rules that would help them understand the problems, and producing solutions to their problems. They found that students “were able to develop progressively more complex forms of mechanistic explanations of emergence” (Sengupta et al., 2018, p. 19) as they worked through the embodied modeling in the classroom. They believed that the emergence of computational modeling, and embodied modeling, in the classroom could help students grasp deeper STEM concepts, and it was their articulation that spurred the authors to create this article.

**Teacher Modeling**

Separate from computational modeling, the first author envisioned modeling as entirely different when it was mentioned in the PD conversation. The foundation for his modeling concept is explicated in Bandura’s Social Learning theory (Tracey & Morrow, 2012). In Bandura’s four phases of observational learning, the students see and watch the model, think about and process what took place, attempt the activity that was modeled, and then receive feedback based on their attempts (Tracey & Morrow, 2012). Coming from a literacy education perspective, the first author envisioned modeling where the teacher *models* the strategy or technique that a teacher wants a student to use (Duke et al., 2011). The authors refer to this as teacher modeling practice in Figure 1. This is when teachers make their invisible thinking about a text visible to the student by doing a think aloud or walking them through a step in a comprehension or composition process (Beers, 2003). Further, Burkins and Yaris (2019) clearly explained modeling as:

> The teacher presents students with a new skill or strategy. At the beginning of the lesson, the teacher provides students a model for what new learning looks like by thinking aloud or otherwise demonstrating the skill or strategy in action. Students watch carefully as the teacher demonstrates. (p. 1).

In other words, showing students teacher thinking, as they demonstrate for the students what it is that teachers do, as they read texts so the students can understand that it is a process that takes place, and not simply something that happens.

**Theoretical Frameworks**

Three specific theoretical frameworks informed this examination of the potential PD modeling disconnect between the faculty and the participants and are explained in more detail in the next sections: terministic screens, schema theory, and theory of capital. Each framework added a level of understanding as to why the participants might have interacted with the discourse differently than the faculty intended.

**Terministic Screens**

Burke (1966) utilized terministic screens to explain how different people understand the same word. A terministic screen is “any nomenclature [that] necessarily directs the attention into some channels rather than others” (p. 45). In other words, the way speakers use words leads listeners to a certain termination, a certain end, which then crucially impacts how that word is perceived after it is used and could potentially produce differences that make understanding more difficult. In a PD, the terms that the faculty used functioned as certain terministic screens that were relevant to what was happening. The participants, on the other hand, may have had a different...
concept of that terministic screen, which could have sent them down a path that was different than originally intended. One term, such as modeling, could have a positive or negative impact depending on the screens through which those terms were understood. What may be a carefully chosen word by the faculty could be viewed through an entirely different terministic screen by the participants, leading to a miscommunication, or worse, and could seriously impact the overall quality of the PD session simply because the terministic screens functioned distinctly for the faculty and participants.

**Schema Theory**

While terministic screens function in both the faculty and participants, schema theory explains how the participants take what is being taught and create their own knowledge and learning. Schema theory suggests that learners attach information to previously generated, though mutable, schema in their own understandings (Tracey & Morrow, 2012). From this perspective, a learner, when confronted with new information, attempts to find connections to preexisting schema in order to make sense of it. When the new information can connect to previous schema, the learner has an easier time learning and understanding the new information. However, if there are no connections, or misleading connections, the learner may have a more difficult time with the new information. Granted, schema can change, and be altered, and information that does not connect to current schema can shift schema for the learner (Tracey & Morrow, 2012), but that initial moment of learning can be either positively or negatively impacted by the connections to the schema of the learner. As an example, Quinlan (2019) utilized schema theory as a framework to explain what happened as preservice teachers engaged in digitally examining an authentic crime scene. She argued that “schema development would be akin to talent development that is sustained and becomes a working schema that students could retrieve and apply to other situations” (Quinlan, 2019, p. 417). In the case of this research study, the modeling question was not just about application of working schema, but about how that working schema of the participants might be different than that of the faculty.

Terministic screens and schema theory functioned together throughout this study. The second author was excited about using modeling in the PD and spoke enthusiastically about the potential for integrations across K-12 classrooms, supports that it could offer students, and the potential benefits for teaching integrated STEM content. Yet, the terministic screen for modeling was the stumbling block. The second author referred to computer science modeling of scenarios (Sengupta et al., 2018); the first author referred to the teacher practice of I do-we do-you do (Burkins & Yaris, 2019). From there the schema of the first author struggled both to understand the second author and other faculty, and to more fully understand how modeling was going to be used in the classroom. The terministic screen caused the initial confusion, but the first author also had no schema to attach to computer science modeling. This initial miscommunication, based on terministic screens and schema, foreshadowed the difficulties of the participants in the RAMPED PD sessions.

**Bourdieu’s Theory of Capital**

The final theoretical perspective assists in examining the motivation for participating in the PD. Bourdieu’s (1986) concepts of capital help to explore the motivations behind those presenting (e.g., faculty) and those attending (e.g., participants) a PD, specifically embodied cultural capital and economic capital. Embodied cultural capital exists within the individual. Bourdieu (1986) wrote, “The accumulation of cultural capital in the embodied state. . . presupposes a process of embodiment, incorporation, which, insofar as it implies a labor of inculcation and assimilation, costs time, time which must be invested personally by the investor” (p. 244). In essence, this is “work on oneself” (Bourdieu 1986, p. 244). As long as the individual has the time to devote to gaining the embodied cultural capital, it can be acquired. Economic capital, on the other hand, can be “immediately and directly convertible into money and may be institutionalized in the form of property rights” (Bourdieu, 1986, p. 243). For the individual, economic capital leads to enhancing wealth, by at least some degree.

Howard (2011), reflecting on plagiarism in writing classes, made the connection between Bourdieu’s theory and the classroom. In her examination, writing instructors viewed the classes and assignments as a way for the students to build embodied cultural capital as they gained the skills necessary for writing in the academy and beyond. However, she argued that “the educational institution is a primary site for transmitting cultural capital, and that cultural capital, once acquired, becomes a tool for acquiring economic capital” (Howard, 2011, p. 223). The difference was between the instructor’s concept of embodied cultural capital as its own end, and the students’ view of the embodied cultural capital as something to convert into economic capital. For the students, efficiency is important, because the goal is conversion; for the instructors, learning is important, since it will enhance the individual’s embodied cultural capital. These concepts of capital can be translated to the study and project described in this article.

Hence, these three theoretical perspectives shaped the analysis and assisted us in understanding why certain aspects of the PD did not match the project RAMPED team’s expectations. Through these theories, the team could see where terministic screens led to issues in participants’ schema; the team also found that participants’ motivation for being in the PD may not have been what was assumed because the faculty and participants were often operating from two different concepts of capital.

**Methods**

During the summer of 2018, teachers from across the west came to a university campus to participate in a 2-week PD
centered around computer science and its integration into the classroom called project RAMPED. Participants attended four sessions throughout the 2 weeks to experience how to integrate computer science into the classrooms and help students identify common computer science themes (e.g., patterns and problem-solving). This PD came at a time when the state placed an increased focus on computer science, mandating that every district teach computer science within the next few years. Specifically, the project focused on meaningful examples that teachers could use to integrate computer science and computational thinking into their classrooms by providing K-12 students real-world instances and hands-on experiences. Importantly for the PD team, the project did not provide a packet of lessons for use, focusing instead on already created lessons freely available and helping participants investigate the underlying computer science concepts and their ability to integrate into other subjects, so they could apply computer science in their own classrooms.

**Participants**

In this qualitative research study, 40 teachers from 19 school districts across the region attended, ranging from elementary teachers to high school teachers (full K-12 range participation). Most participants taught math or science, but there were also teachers of English, choir, and other disciplines, representing the integration that was one of the focal points of the PD (see Table 1). Participants attended four sessions including machine learning, anomaly detection, large astronomical distances, and virtual reality, which all related to computer science and its integrative potential.

**Data**

At the beginning of each session, every participant completed a survey on a number of topics relevant to computer science in the classroom. For this study, two questions were uniquely important. First was, “When you think about modeling in the classroom, what do you envision? What comes to your mind? What do you see?” Second was, “Write about a way that you have used modeling in your classroom or have seen (or heard about) another teacher use modeling in the classroom.” Participants were given no context for these questions, other than that they were for the the project. At the end of the session, participants completed a survey with the same two questions. Again, they were given no instructions, but this occurred after they completed sessions centered around computer science and modeling. It is important to note that computer science in this project included some coding, but it was made explicit that computer science was not coding, and coding was a tool that computer scientists utilized. Additionally, during the sessions over the 2 weeks, 14 participants were interviewed (see Table 1).

These participants were asked to come outside during the day’s activities for the interview as to make the interviews the least disruptive to the teachers’ time. They were offered different days and times to interview and could change if their selected day and time was inconvenient when it arrived. Most interviews were individual; however, two sets of participants interviewed together due to time and scheduling constraints. The questions revolved around computer science and coding (both familiarity with and use in the classroom), STEM integration, and modeling. Interviews were recorded and lasted between 10 and 20 minutes. Finally, during the following academic year (2018–2019), participants were encouraged to apply what they learned, and at the end of the year, they responded to a final survey which asked them what they incorporated into their classrooms, how often they incorporated new ideas from the PD, and what the outcome was after their yearlong experience.

**Analysis**

All data points were coded using a process called themeing the data (Saldaña, 2013). Because modeling was one focus of the project PD, as well as one of the initial confusions between the authors, the data was themed by looking for instances when the participants defined or described modeling. At those points, the data were coded as “modeling is x.” For example, one participant said, “I envision modeling as me modeling what they need to do, and so the students watch me do it, and then they do it.” This was coded as, “modeling is I do-we do-you do.” In response to the same question another participant said, “I think about programmatic representations of any interesting system or process. I mainly use agent-based modeling with students because they enjoy it.” This was coded as “modeling is a computer model.” All interviews, pre- and post-surveys, and emails were coded in a similar fashion. On the first coding, eight themes were developed throughout the data, and then applied to each. For the second coding, those eight initial themes were combined into three meta-themes, and all data were coded again with these meta themes to ensure that all categories were used for each data point. Once coding was complete, examples of each code were then analyzed to look for common trends or ideas throughout the data.

**Findings**

One participant clearly articulated the main problem that the authors found in the data: “I’m taking this computer class, and he keeps using this word “modeling” in a different term. . . so I don’t think he’s thinking the same modeling. I think we’re not on the same page.” This discrepancy
continued to appear; even though some participants did understand the faculty’s use of modeling, the majority did not.

**Surveys**

In the pre surveys, responses to the modeling question were grouped into eight initial themes, and then grouped into three meta themes: modeling is computer/conceptual modeling; modeling is I do-we do-you do; modeling is projects. There were 5 instances of participants writing that modeling is computer/conceptual modeling, 29 instances for I do-we do-you do, and 6 instances for modeling as projects (see Figure 1). The difference between total participants and total coded instances is rooted in that some participants described modeling as more than one meta theme. Given the lack of context before participants responded to this survey, their responses were not surprising, but it did show that there were differences in the terministic screen modeling that could lead to differences in schema access and understanding. Because of

---

### Table 1. Participants in project RAMPED.

| Gender | Teaching level     | Content area            |
|--------|--------------------|-------------------------|
| Female | Elementary school  | Computer science        |
| Female | Elementary school  | Technology/library      |
| Male   | Middle school      | Math/science            |
| Female | Middle school, high school | Math/science         |
| Female | High school        | Social studies          |
| Female | K-12               | Library                 |
| Female | Elementary, middle school | STEM classes    |
| Female | Middle school      | Engineering             |
| Female | Middle school      | Engineering             |
| Male   | Middle school      | Technology/library      |
| Male   | High school        | Industrial arts         |
| Male   | High school        | Computer                |
| Male   | Middle school, high school | Choir              |
| Female | Elementary school  | First grade             |
| Female | Elementary school  | Computer lab            |
| Female | High school        | Science                 |
| Female | High school        | Special education       |
| Male   | High school        | Science                 |
| Female | High school        | Science                 |
| Female | Elementary school  | Multiple grades         |
| Female | Middle school      | Pre-service science     |
| Male   | Trade school AAS. Business | Welding           |
| Male   | High school        | Welding                 |
| Male   | Elementary school  | Physical education      |
| Male   | High school        | Science                 |
| Female | K-12               | Library                 |
| Male   | Elementary school  | Kindergarten            |
| Female | Elementary school  | Third grade             |
| Female | High school        | English                 |
| Female | Middle school      | Math/science            |
| Female | Elementary school  | Fourth grade            |
| Female | Elementary school  | Second grade            |
| Female | High school        | Science                 |
| Female | Elementary school  | First grade             |
| Male   | Middle school      | Science                 |
| Female | Elementary school  | Science                 |
| Female | Middle school      | Math                    |
| Female | High school        | Science                 |
| Male   | Elementary school  | STEM classes            |
| Male   | Elementary, middle school | Science           |

*Participants were interviewed during the 2-week session.*
the numbers of coded instances, the first two meta-themes will be further analyzed.

Those who wrote of computer/computational modeling tended to write something like this participant: “When I think of modeling, I think of systems and physical or mathematical representations of real-world phenomena. I see student/student and student/teacher collaboration to define system boundaries and determine appropriated scaling, variables, and constraints.” There was a clear focus on using the model to represent a system of processes so that students could see or understand how something might function. For example, another participant wrote, “The kids were studying wildlife issues in [state] and they created models in NetLogo to show how some of the pressures [that] wildlife face change their breeding, eating, etc. . . patterns.” This response closely aligns to the perspective of the faculty of the PD.

However, even though the I do-we do-you do was never brought up as a model, those who thought of modeling as I do-we do-you do represented the majority of the participants. One participant was very succinct: “When I think of modeling in the classroom, I think of the classic approach of I do, we do, you do. This approach can be used in any situation with teaching be it modeling behavior, reading strategies, math games, basic robotics kits like WEDO2.0, and art projects.” In this case, modeling was not limited to computer science, which was one facet of the project; in fact, the participant noted that it could be used in any situation. Another participant linked the concept to computer science but kept the concept of the model the same: “I think of the I do-we do- you do approach. I’m not in a regular classroom. I’m strictly in the computer lab! When it comes to programs, Lego robotics or daily lessons, it’s always best to show and gather understanding together through hands on approach before turning [the students] loose.” Modeling was teaching the students how to do the procedure, demonstrating for them, and then working with them as they moved into the process themselves. Again, even though this participant was in a computer lab, the concept of modeling dealt with helping students perform a task and not what the faculty envisioned with the concept of modeling.

The post surveys were completed in a classroom on the last day of the PD. Despite 2 weeks of intensive classes on computer science modeling, the results on the survey were unexpected. There were 24 instances of defining modeling as I do-we do-you do (down from 29 pre-survey responses), compared to only 11 instances of modeling as computer/conceptual modeling (up from five pre-survey responses), despite the continued use of modeling as computer/conceptual modeling by the faculty. There were also four instances of modeling as projects (down from five pre-survey responses). Even though this survey was completed during the last session, four participants did not complete the final survey.

As a PD success, the 11 instances of modeling as computer/conceptual modeling was more than the pre survey, and those who viewed modeling as computer/conceptual modeling showed that they matched the faculty members’ concepts. One participant wrote, “Modeling is a representation of evidence, or novel approaches to visualized problem/solutions/inquiry.” Another wrote that modeling was “creating situations – physically or on a computer – that allows students to conceptualize big pictures into meaningful pieces.” One participant wrote that modeling was “building a system to perform a task” and “anomaly detection.” All of these definitions were part of the 2-week session.

Despite the focused PD, though, the dominant concept of modeling was still I do-we do-you do. Participants tended to phrase it exactly like that, as this participant did: “I do, we do, you do. I like to give a ton of examples to make sure everyone can be independent with their learning.” And another one: “I do, you do, we do. Students learning by observing instructor and peers and then being able to apply and retry.” Some thought that modeling was just demonstration, the I do part of the modeling: “I envision demonstration. I demonstrate proper tone production, etc., all the time.” Another participant wrote, “Teacher (usually) demonstrating a skill, procedure, or finished project.” This concept of modeling fit with the I do-we do-you do structure; they just wrote about the first part without referencing the second and third parts.

**Interviews**

The interviews, which were conducted during the course of the 2-week sessions, yielded similar results, with one exception. No one interviewed spoke about modeling as projects. However, there were 21 coded instances of modeling as I do-we do-you do, and 11 coded instances of modeling as computer/computational modeling. Even during the PD, participants kept their mindset about what modeling was and tried to fit the project pieces into that mindset.

Participants who spoke of computational/computer modeling said similar things. One said, “You’re talking about vast systems of looking at large amounts of data and running it through processes and getting trends out of it.” Another participant was more specific:

Let’s say you’re doing like the Ideal Gas Law. So, you can start to manipulate these variables in a simulation, and so as you manipulate these variables you can start to develop an understanding of the patterns and trends and when this goes up, this goes down. When this – so I see it as a way to help students make connections with more complex ideas.

In these cases, the participants understood the focus of the PD, and began either to modify their concepts of modeling, or to confirm that they were on the right track.
However, many participants still operated under the concept of modeling as I do-we do-you do. “So, modeling for me is when it’s I do, we do, you do. So, the instructor or the teacher demonstrates the expectation. Then we all go through the expectation together. And then the [students] are left to do it on their own.” Another participant said, “There’s modeling where I go up and show them how something is actually done and then I expect them to come back and do it in that fashion or do something similar and take it a step further.” Even when they spoke of computer science ideas, some still referred back to the same concept:

I like to take the [students] maybe step-by-step the first time, teach them the basic concept, basic programming, basic drag-and-drop and then kind of let them explore a little bit and make combinations for themselves and let me – let them show me what they have learned or what they’ve done. And then I’ll come back and teach another piece. . .

The initial concept of modeling, which was present before the session began, continued to be a dominant concept even during and after the 2-week PD.

Email Reflections

Overall, 25 participants responded to the email questions sent out at the end of the yearlong PD and follow-up sessions. All but two participants reported using something new from the summer PD; however, the addition ranged from a single 15-minute lesson to a weekly 1-hour activity time. The average amount of new lessons for the group was approximately five lessons or class periods for the year.

The discrepancy between participants was echoed in their comments as participants reported a range of impact on their professional practices from transformation to unengaged. For example, one participant wrote:

...computer science is about problem solving and critical thinking and it enhances what we are already doing in schools. It took me all of RAMPED to understand this concept. . . .we have evolved from finding the “right answer” to creating solutions.

Another wrote, “I purposely inject content from RAMPED into class discussions such as data collection, anomaly detection, algorithms, and spatial analysis.” These are representative of the participants who had positive interactions with the content. Those who had less engaged reactions wrote of the inability to add PD content because of time, budget, or supply constraints. A few, though, wrote about the inherent language problems that created a disconnect between the faculty and the participants. One participant wrote:

Even though all the [faculty] presented their topics very well and demonstrated great knowledge and willingness to enlighten us, they were operating at a level of comprehension several leagues above your average public-school teacher. I was often lost on much of the math and programming language. Sort of like listening to the air traffic control talking to aircraft coming into a large airport.

This participant was complementary and positive about the event, and attempted to implement some of the lessons, but the overall tone was one of frustration and a lack of comprehension. Another wrote, “...everything we did in the summer class was at too high of a level from me to understand, let alone teach to 3rd graders.” In both of these cases, the language used, and the level of math discussed, created comprehension problems for the participants that made it difficult to bring the concepts to their classroom, despite their stated desire to do so, and despite the efforts of RAMPED faculty to explicitly bring basic concepts such as patterns and problem-solving into every session.

Time to create lessons, and ability to use those lessons as part of a district curriculum also posed problems for some participants. Although it was explicit that lessons should tie into already existing curriculum needs, one participant wrote:

I think it would be beneficial to create lessons that tie into our curriculum; being able to adjust some of the language of the lessons we are required to teach, to integrate computer science ideas would be much more helpful than lessons I have to find time to teach separately, make meaningful, and connect to something. . . .

As stated above, the goal of project RAMPED was not to provide a set of lessons, but to create a deeper understanding for computer science application. For this participant, however, the lessons learned from the PD felt more like extracurricular subjects in a school curriculum that was already defined. This example, which was echoed by some other participants, demonstrates some of the differences in transferable lessons between the faculty and participants. The faculty wanted to change a mind set about computer science and modeling in the classroom; some participants wanted immediate lessons to instantly enact in the classroom.

Surprisingly, no participants mentioned modeling specifically, either the computational modeling that was the focus of the project PD, or the I do-we do-you do modeling that was prominent in the participants' conceptual understanding of the term. This could be due to the lack of a specific modeling question on the email survey, or in the implicit nature of the modeling definition during the session, but it could also be because the participants had returned to their home schools, and their personal terministic screens and schema, and did not see a need to explain it further. Either way, the absence of modeling from the email responses was curious.

Discussion

As educational researchers the knowledge that strong teacher PDs are needed and that changing mindsets is difficult are givens, however, even though those two pieces were tackled
during the PD creation, there were persistent disconnects. There is great positive potential for computer science modeling in the K-12 classroom. Sengupta et al. (2018) argued “by engaging in iterative cycles of building, sharing, refining, and verifying computational models, students refine their understanding of what actions and interactions of agents represent an “event,” which are then displayed on graphs” (p. 17). This promise is what drove the heart of the RAMPED project. By helping participants visualize and understand this potential, the faculty hoped those participants could take the learning back to their home classrooms to engage their students in new levels of computer science thinking. However, during the sessions, and in subsequent check-ins with the participants, the project team found a divide that was keeping the positive messages about the computer science modeling practices from being fully realized by the participants. As a result, the team asked if it was the language used by the instructors, and the interpretations of that language by the participants, that created the issues.

The authors found that this was the case, especially with the term model. The majority of the participants came to the PD with a terministic screen of model firmly in place. When participants heard model, they immediately thought of the I do-we do-you do concept, as evidenced by their responses in the interviews and surveys. In fact, it appeared that despite the instruction on modeling throughout the session, participants did not move past their terministic screen of modeling. Consequently, the computer science modeling that the faculty were teaching had no schema to attach to in the participants’ views, which led to the disconnect. When they heard model and attached that to their schema of model, their attention was necessarily directed toward certain things (how the teacher models, what the model is, how the students learn from the model) and not toward what the computer science model taught by the instructors represented. Even though it is just one word, the terministic screen for model and its attached schema were powerful enough to overcome even direct instruction by the faculty in each session.

In this case, though, there was even more to cause some of the disruption of understanding. Participants had the opportunity to take this PD for certification credit, and to receive a small stipend. In other words, there was an economic capital readily available for them. All they needed to do was attend, try something during the year, and report back. A certain amount of certification credit is necessary to maintain teacher licensure in the state, which is necessary for the participants to continue teaching. By participating in project RAMPED, they earned this economic capital, both credit and monetary, and continue with their profession. This did mean that there was a different level of buy-in on the part of the participant. The instructors viewed this time as building embodied cultural capital. They wanted participants to change the way they viewed computer science in the K-12 classroom. This PD was a method for the faculty to assist the participants in developing their own deeper understandings of the topics which would, in turn, help them become better teachers, and help their students to experience new kinds of thinking in relation to computer science.

During the summer, as participants were engaged in the sessions and separated from their home schools, there seemed to be a building of the embodied cultural capital. Participants were positive about the sessions, about the learning, and were willing to spread that positivity in the interviews, post surveys, and in the email responses. For example, in one email response, the participant wrote, “Each month I think I become more confident and open to computer science and that will simply benefit my students. It has been a great journey!” However, once the session was over, and teachers went back to their home schools, the economic capital become prominent, especially as they dealt with a lack of time and resources to do in their schools what they wanted to do. Enthusiasm built up over the 2 weeks dwindled to one participant’s comment that “nothing in the summer class really helped me prepare for this at the elementary level.”

By the time the participants began teaching in the next school year, the economic capital became more predominant, and colored their desire to apply what they learned, to the point that they wanted pre-made lessons instead of crafting their own. They fulfilled their obligations to receive that economic capital, and then moved on. This, also, contributed to the second author’s feeling upon reflection. No teachers emailed or asked for follow up or extension help throughout the year. In fact, the team only heard from the participants when the email was sent asking for them. The desire to help teachers enhance their own embodied cultural capital was unmet, overwhelmed by the economic capital that teachers received upon completion.

The project team does recognize several limitations in this study. One limitation is attempting to reach a large audience (40 teachers) of all subjects and grade levels. This was an intentional choice by the project team (which included state leaders) as a proof of concept. The team realized that past research shows that this is not the best method to hone PD messages and experiences, and this study reinforces that idea (e.g., elementary teachers experience different needs than secondary teachers). Another limitation was the distance between participants and faculty. While the participants and faculty were mostly from one state, the geographical distances between the individuals ranged between a few to hundreds of miles. Although email and other virtual meeting mechanisms were in place, this distance proved troublesome to some participants that wanted more one on one instruction. The third large limitation was the self-selection of the teachers to participate in the PD. This was not a random sample of participants.

**Implications**

As the team planned and enacted the next PD, there were two major implications that were utilized for computer science
modeling. First, the team did not take terms for granted, nor did they assume that because the content had been announced and because the term was a vital part of that content, that participants would necessarily gain this knowledge without explicit definitions of the terms and a focus on schema building. The team should have noticed this earlier as the authors spent a few weeks each thinking about modeling but doing so in completely different ways. By the time the authors came to this understanding, it was too late for replanning sessions, and the authors, from different disciplines, thought it was only a miscommunication. The authors, after the PD, found mismatched mindsets for modeling, integrated STEM, and even for simple messages that were made explicit like problem-solving, critical thinking, and patterns.

The project team encourages being explicit about the definitions and concepts being used, not just in how they are used, but also in how participants demonstrate and apply them. This may sound simplistic in educational settings. However, content areas and cultural interactions are different enough that the same terms are used in multiple content areas, but with different meanings. For example, in the Common Core State Standards (CCSS) for mathematics, computational modeling is present, and aligns with what the faculty believed about modeling. However, in writing instruction, modeling refers to both the teacher writing along with the students and presenting the model, and with teachers using examples, or models, from the texts that students are reading to help them explore new writing techniques. Elementary teachers, like the ones who attended the PD, can be exposed to one or both of these definitions. If PD providers do not clearly and consistently explain and define the disciplinary terms used, and give students the opportunity to demonstrate their understanding as presented, then PDs run the risk of opposing participants’ terministic screens, and either lack of schema or misplaced schema.

Second, the team cannot underestimate, nor demean, the participants’ desire for economic capital as a result of attending the PD session. In fact, the team acknowledged it and worked with the teachers to embrace embodied cultural capital as well as economic capital, but not explicitly. In PD settings, the providers should admit that the participants might be there for a different reason than learning. The participants may need credit, money, or they may have been directed to attend. In these cases, the economic capital is the motivating factor. If the PD providers embrace that idea, then it can be clear what they need to do to earn the economic capital they desire as well as what they need to do to engage in new mindsets and experiences. Once economic capital is acknowledged then PD providers can focus on the embodied cultural capital. This takes a potentially hidden secret in the room—economic capital motivation—and displays it, allowing everyone to see it and to discuss it. Once that weight is lifted and becomes part of the discourse, all involved can focus on the personal improvement and growth at hand.

These may appear to be inconsequential steps, too small to be worthwhile and too obvious to be mentioned. However, the absence of one or both of these can lead to either a derailing of the PD, or at least a feeling of failure by those who planned the event. In this case, the team’s expectation of teachers producing sustained computer science modeling in their classrooms was not consistently met. During the 2 weeks, everyone was engaged, integrated STEM activities were explored, and participants showcased their work. The interviews were full of positivity. Even the post surveys and email responses had positive things to say about the experience. However, the transfer to practice was not completely present, despite the instructors’ efforts to work with the teachers on their schedules and enact any part of the program they desired. A change in explicitly defining the terms, and embracing the competing motivations, though, may have had an impact. Lastly, the authors, wanting to explore the successes as well as the K-12 teacher challenges, write this article to open discussion around computer science education and modeling, and encourage others to continued research studies on language use and STEM integration.

Acknowledgments
The PD team would like to thank the teacher participants who worked with us in creating and implementing the RAMPED PD and then creating an improved PD model upon reflection for SWARMS and WySLICE.

Declaration of Conflicting Interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Wyoming Department of Education (WDE) and the National Science Foundation (NSF) through the Sustaining Wyoming’s Advancing Reach in Mathematics and Science (DUE Award #1339853) and through WySLICE the CS for All: RPP—Booting Up Computer Science in Wyoming (DRL Award #1923542). Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF

Ethical Approval
This STEM PD study was approved by the Institutional Review Board of the University of Wyoming (protocol #20150921AB00901, 21 September 2015).

Informed Consent
Written consent was obtained from the participants.

ORCID iD
Andrea C. Burrows (https://orcid.org/0000-0001-5925-3596
References

Abrahams, I., Reiss, M. J., & Sharpe, R. (2014). The impact of the “getting practical: Improving practice work in science” continuing professional development programme on teachers’ ideas and practice in science and practical work. Research in Science and Technological Education, 32, 263–280.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? ACM Inroads, 2(1), 48–54.

Beers, K. (2003). When kids can’t read what teachers can do: A guide for teachers 6–12. Heimann.

Bell, T., Urbahne, D., Schanze, S., & Ploetzner, R. (2010). Integrating computer science into teacher professional development with NetLogo. Education Sciences, 9(1), 66.

Bourdieu, P. (1986). The forms of capital. In J. G. Richardson (Ed.), Handbook of theory and research for the sociology of education (pp. 241–258). Greenwood Press.

Burke, K. (1966). Language as symbolic action. University of California Press.

Burkins, J., & Yaris, K. (2019). Next generation scaffolding and gradual release of responsibility: Quick reference guide. NCTE.

Burrows, A. C. (2015). Partnerships: A systemic study of two professional developments with university faculty and K-12 teachers of science, technology, engineering, and mathematics. Problems of Education in the 21st Century, 65(1), 28–38.

Burrows, A. C., DiPompeo, M. A., Myers, A. D., Hickox, R. C., Borowczak, M., French, D. A., & Schwortz, A. C. (2016). Authentic science experiences: Pre-collegiate science educators’ successes and challenges during professional development. Problems of Education in the 21st Century, 70, 59.

Duke, N. K., Pearson, P. D., Strachan, S. L., & Billman, A. K. (2011). Essential elements of fostering and teaching reading comprehension. In S. Samuels & A. Farstrup (Eds.), What research has to say about reading instruction (4th ed., pp. 51–93). International Reading Association.

Faulkner, J., & Latham, G. (2016). Adventurous lives: Teacher qualities for 21st century learning. Australian Journal of Teacher Education, 41(4), 137–150. https://doi.org/10.14221/ajte.2016v41n4.9

Garet, M. S., Porter, A. C., Desimone, L., Birman, B. F., & Yoon, K. S. (2001). What makes professional development effective? Results from a national sample of teachers. American Educational Research Journal, 38, 915–945.

Grover, S., & Pea, R. (2018). Computational thinking: A competency whose time has come. In S. Sentance, E. Barendsen, & C. Schulte (Eds.), Computer science education: Perspectives on teaching and learning in school (pp. 19–38). Bloomsbury Academic.

Howard, R. M. (2011). Postpedagogical reflections on plagiarism and capital. In S. I. Dobrin, J. A. Rice, & M. Vastola (Eds.), Beyond postprocess (pp. 219–231). Utah State University Press.

Johnson, C. C., Mohr-Schroeder, M. J., Moore, T. J., & English, L. D. (Eds.). (2020). Handbook of research on STEM education. Routledge.