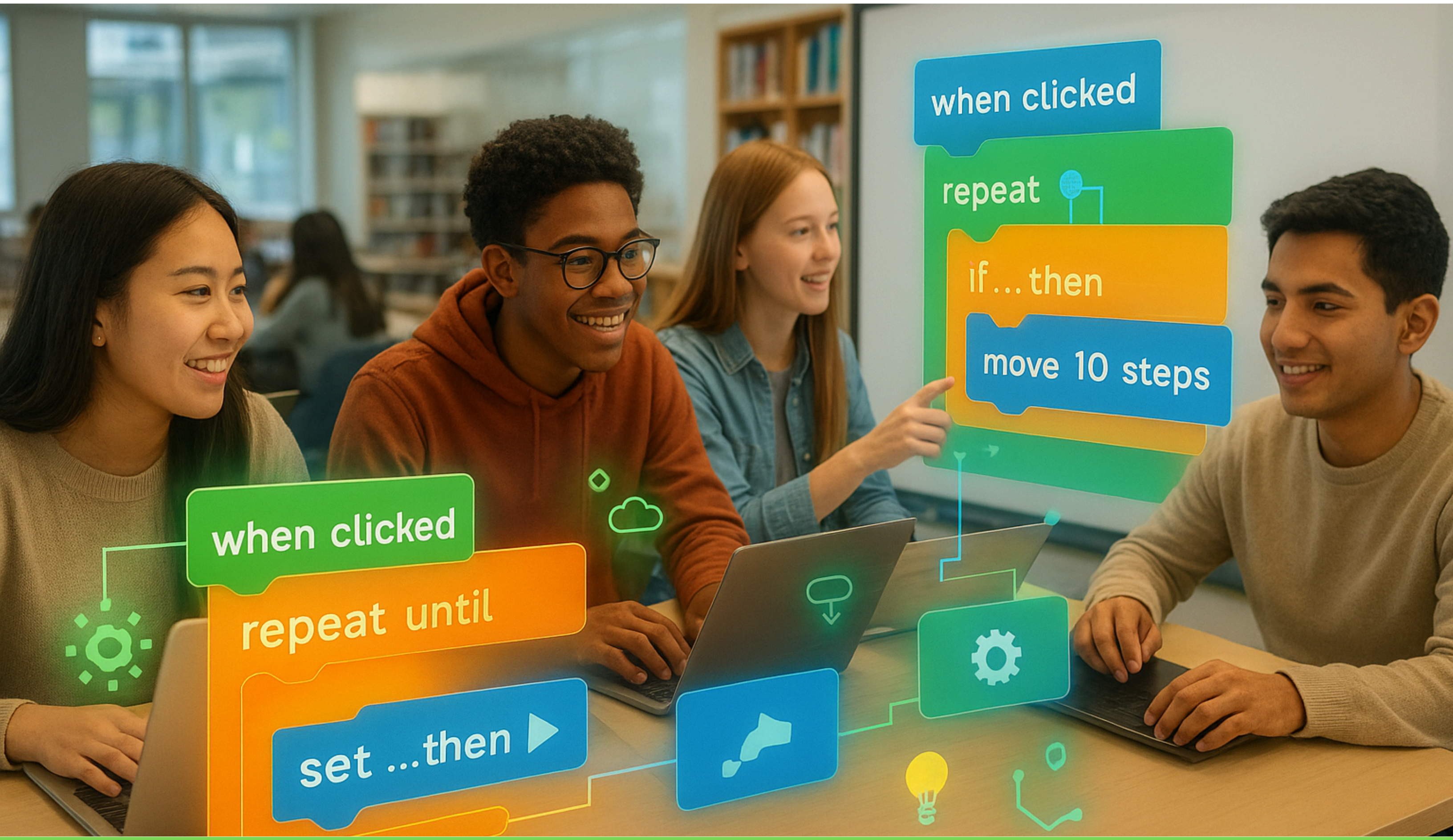
Computer Science Integration

Enhancing Math and Science Instruction with Professional Learning and Model Lessons



SSDA Small School Districts' Association | CS4NorCal | SCOE Sacramento County Office of Education

Computer Science Integration

Enhancing Math and Science Instruction with PL & Model Lessons

Lead Agency:	Small School Districts' Association (SSDA) 925 L Street, Suite 1185 Sacramento, CA 95814
Project Directors:	Kathy Hamilton Karen Mix
Author:	Jonathan Rhodea
Editor:	Jared Amalong
Producer:	Jessica Heidrich
Contributors:	Garrett Edwards
Contacts:	Karen Mix - karen@ssda.org Jonathan Rhodea - jrhodea@scoe.net
Funder:	U.S. Department of Education Office of Elementary and Secondary Education Education, Innovation, and Research (EIR) Grant Program
EIR Cohort Year:	2019
Grant Period:	October 1, 2019, to September 30, 2025
Award Number:	U411C190234

Funding for this report was provided through an Education Innovation and Research (EIR) grant from the U.S. Department of Education, Office of Elementary and Secondary Education. The grant was awarded to the Small School Districts' Association (SSDA), with professional learning services provided by the Sacramento County Office of Education (SCOE). The opinions, findings, and conclusions expressed in this report do not necessarily represent the official positions or policies of the funding agency. Correspondence regarding this document should be directed to the Sacramento County Office of Education, P.O. Box 269003, Sacramento, CA 95826-9003.

Table of Contents

Abstract 4
Part I - Background and Context 5
Introduction 5
Targeted Literature Review 6
Integrating CS into Core Subjects 6
Connecting CS to Local STEM Careers 6
Employing Engaging and Accessible Tools 7
Conceptual Framework 7
Part II - Student-Facing Model Lesson Plans 9
Product Design & Development 9
Computing Systems 10
Networks and the Internet 10
Algorithms and Programming 10
Data and Analysis 11
Impacts of Computing 12
Implementation & Fidelity 13
Student Outcomes 14
Part III - Professional Learning Component for Educators..........15
Product Design & Development 15
Implementation & Fidelity 16
Educator Outcomes 17
Part IV - Next Steps and Closing 18
Discussion 18
Limitations 19
Future Research 20
Conclusion 21
References 22
Appendices 23
Appendix A - CS4NorCal Logic Model 23
Appendix B - Lesson Plans 24
Appendix C - Professional Learning Tollkit 25

The contents of this report were developed under a grant from the U.S. Department of Education, Education Innovation and Research (EIR) Program. However, those contents do not necessarily represent the policy of the U.S. Department of Education, and you should not assume endorsement by the federal government.

Abstract

California's small, rural school districts face significant systemic barriers to providing equitable computer science (CS) education, including funding inequities and limited access to professional development. To address this disparity, the CS4NorCal project, funded by a federal grant to the Small School Districts' Association, developed a regional, context-responsive model to build local capacity and create sustainable CS pathways. A key project strategy is the integration of computer science concepts within the core curriculum, which CS4NorCal addresses through CS Integration (CSI), a two-pronged approach featuring both hands-on lessons and professional development.

The first component of CSI is a suite of student-facing model lessons, aligned to state standards and structured around the 5E instructional model, that used hands-on physical computing to make abstract CS concepts tangible. The second is an intensive professional learning program that equips educators with the pedagogical strategies needed to effectively implement these lessons and design their own.

For professional learning practitioners and school leaders, this article examines the CSI model as a case study in designing and adapting curriculum and training for a specific, high-need context. It highlights key outcomes, including a significant increase in educator confidence to teach CS and consistent positive student feedback from a pilot study. Furthermore, it connects the CSI model to the broader project's statistically significant positive effect on student science achievement. By grounding the model in inquiry-based pedagogy and accessible, hands-on tools, this work offers a promising and replicable framework for expanding equitable access to CS education.

CS4NorCal
www.cs4norcal.org

Part I - Background and Context

In 2018, California established statewide computer science (CS) standards but failed to provide a clear implementation plan for its numerous small, rural school districts. These districts face systemic barriers, including funding inequities, insufficient technological infrastructure, and limited access to professional development, that have impeded their ability to offer the same opportunities as their urban peers. During the 2019-2020 school year, only 24% of California's rural high schools offered a CS course, compared to 56% of schools in suburban and urban areas (Code.org et al., 2020).

One in every ten students in California lives in a rural area—over half a million students—and 67% of the state's school districts are classified as small (Jones, 2019).

The reasons for this gap are systemic. A national report found that superintendents in rural districts are significantly less likely than their urban counterparts to agree that their school board is committed to offering CS (Google & Gallup, 2020). These barriers include funding inequities, insufficient technological infrastructure, and limited access to professional development.

"Why not here? Why should our kids be behind the 8-ball because of where they live?"
– Rob Adams, retired Superintendent, Redding Elementary School District

To address this disparity, the Small School Districts' Association (SSDA) secured a nearly $4 million federal grant to launch CS4NorCal. The project focused on six of California's most remote counties, where some districts serve fewer than 100 students, educators may drive hours for training, and unreliable internet can hinder online learning. CS4NorCal designed and delivered professional learning and student-facing resources tailored to overcome these specific challenges, building local capacity to create sustainable CS pathways. One of CS4NorCal's most impactful strategies to expand student access to CS instruction was the development of the CS Integration professional learning series, which equipped secondary math and science teachers with model lessons and hands-on tools to seamlessly embed computer science into their existing curricula.

CS educators collaborating during a workshop

Targeted Literature Review

In response to the need for practical ways to teach state CS standards in rural, under-resourced classrooms, the CS Integration professional learning series was developed from a foundation of established educational research and iterated upon during the duration of the CS4NorCal project.

Integrating CS into Core Subjects

Research underscores the value of integrating computer science into core subjects, especially in regions with limited access to standalone CS courses. A study in Chicago, Illinois, found that students who accessed CS coursework performed better in their science classes (Yen et al., 2018). By integrating computational thinking into existing math and science courses, the project created a "bridge" to CS that bypassed common barriers like funding shortages and a lack of specialized teachers.

This integration model is particularly crucial for the remote counties served by CS4NorCal. Four of these counties are considered "Teacher Education Deserts" with no local teacher preparation programs (UCLA Center for the Transformation of Schools, 2021). Equipping current math and science teachers to embed CS into their lessons is a practical and sustainable strategy for expanding student access in these contexts.

Connecting CS to Local STEM Careers

Research also suggests that rural students are more engaged and successful in STEM education when their learning connects to real-world phenomena and potential careers (Smith, 2012). By contextualizing CS skills, educators can help students align their aptitudes with potentially rewarding career paths (Holland, 1997).

By emphasizing career connections, CSI also directly addressed the "rural brain drain," where students often leave their communities for economic opportunities. The Computer Science Teachers Association (2021) noted that equipping students with in-demand technical skills can help them secure high-paying, remote-friendly jobs, allowing them to build a career without leaving their community.

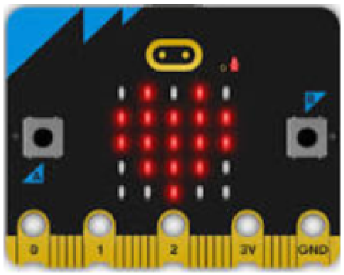
> **A workshop participant appreciated:** "Making the projects. Working with everyone else in the room, trying to figure out the projects. After the projects, debriefing sessions. I like how our presenters showed us lessons using quality teaching practices."
> -CSI participant

Part I - Background and Context

Employing Engaging and Accessible Tools

To further illustrate the CS standards, CSI professional learning and model lessons utilized physical computing. Tools like the BBC micro:bit microcontroller, paired with visual block-based languages like Microsoft MakeCode, lowered the barrier to entry for both teachers and students. By removing the steep learning curve and unforgiving syntax common to text-based coding, students use drag-and-drop blocks to focus on computational logic and problem-solving. This hands-on, project-based approach has been shown to significantly boost student motivation and foster critical thinking skills (Lopez et al., 2016).

Makecode block coding environment



micro:bit microcontroller

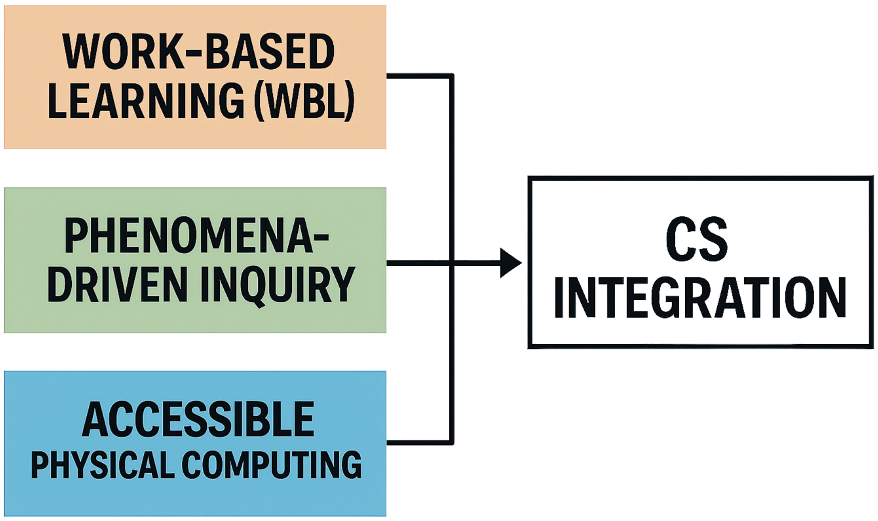
This approach also helps cultivate a positive perception of CS. After using the micro:bit, 85% of teachers reported that CS was more enjoyable for students, and 70% of teachers who were previously hesitant felt more confident teaching the subject (Gibson & Bradley, 2017). The impact on students is equally significant: one study found that using the micro:bit showed 90% of students that anyone can code and inspired 70% more girls to say they would choose to study computer science.

"The material was very practical and lessons I could use in my class right now. This is very valuable and appreciated. I also like that there is ongoing training so that as things come up we can bounce ideas and ask questions to out trainers and other participants."
-CSI participant

Conceptual Framework

Guided by this body of research, CS4NorCal developed a logic model (Appendix A) to translate these evidence-based principles into a concrete implementation plan. The logic model identified specific outcomes - such as increased teacher confidence and student interest in CS -- along with performance targets for delivering professional learning, implementing CS pathways, and providing work-based learning activities.

Part I - Background and Context



The Computer Science (CS) Integration component of the logic model sought to reframe CS not as a standalone subject, but as an essential tool for scientific and mathematical inquiry. This approach is designed to enhance student learning and prepare students for the demands of modern STEM careers. The component's central thesis is that by engaging students in authentic, inquiry-based experiences, educators can simultaneously cultivate deep content knowledge in math and science, as well as crucial computational thinking skills. With that said, CSI is built upon three pillars: Work-Based Learning (WBL), phenomena-driven inquiry, and accessible physical computing.

First, CSI sought to address the persistent gap between how computer science is applied in the professional world and how it is often taught in K-12 education. While professionals use technology as a platform for creative problem-solving, educational institutions often relegate computers to tools for consuming pre-designed software. The CS Integration model empowers students to mirror the practices of scientists and engineers; instead of simply learning about science, students learn to do science with computational tools they design themselves. For instance, students can write code for a microcontroller to measure turbidity in water samples or program a sensor to create an early-warning system for a model earthquake. These activities foster both technical fluency and scientific literacy, equipping students for interdisciplinary challenges.

Beyond its pedagogical strengths, an integrated model offers significant, pragmatic advantages for schools facing resource or logistical constraints, such as those in many rural districts. By embedding CS within existing science and math classes, the framework helps overcome common implementation challenges. This approach avoids complex master schedule changes required for a new standalone course, addresses teacher credentialing hurdles by allowing credentialed science and math teachers to incorporate CS into their existing courses, and promotes equitable access by ensuring all students receive meaningful exposure to computer science within their core STEM education.

Part II - Student-Facing Model Lesson Plans

To translate this theoretical foundation into practice, CSI was designed with a two-pronged approach: developing student-facing model lessons and creating a professional learning program to support educators. Staff at SCOE developed model lessons intended for direct use in classrooms, which also served as examples for educators participating in professional learning. These lessons were created to equip secondary math and science teachers to embed computer science into their existing curricula. The professional learning component introduced these lessons to educators, providing them with resources and strategies to integrate CS activities and showcase new pedagogical approaches. The following section, Part II, delves into the first of these components: the design and development of the student-facing model lesson plans.

Product Design & Development

The CSI model lessons were developed based on a proven pedagogical framework to ensure a high-quality learning experience. To ensure comprehensive and relevant content, all lessons were developed to align with the California K-12 Computer Science Standards. Each lesson was structured around the 5E instructional model, a student-centered, inquiry-based approach that promotes deep conceptual understanding. This section provides an overview of the model lessons, showcasing how they are organized across five core computer science concepts and how the 5E model is implemented.

The 5E Constructionist-Based Lesson Format:

- Engage: This initial phase captures students' interest and encourages them to think about the topic. It's designed to hook the learner, access prior knowledge, and reveal potential misconceptions. Activities might include a compelling question, a surprising demonstration, a video clip, or a short, interesting problem.
- Explore: During this phase, students actively explore the concept through hands-on activities. They work collaboratively, investigate questions, and test ideas. The teacher acts as a facilitator, providing guidance and materials but allowing students to grapple with the content themselves.
- Explain: In this phase, students articulate their understanding of the concepts they just explored. The teacher then provides direct instruction on technical skills and formal definitions for key vocabulary. This phase solidifies learning by connecting the students' hands-on experiences to the academic concepts.
- Elaborate: Students apply what they've learned to new, but similar, situations. This phase deepens their understanding and helps them make connections to real-world scenarios or other academic disciplines.
- Evaluate: Assessment occurs throughout the lesson. Both students and teachers assess understanding and skill acquisition through formative assessments like exit tickets or observations, or summative assessments like projects or quizzes.

Part II - Student-Facing Model Lesson Plans

Additionally, the model lessons were organized sequentially across five core CS concepts:

1. **Computer Systems**

As defined in the California K-12 Computer Science Standards, this concept focuses on the components of a computer (hardware and software) and how they work together.

- Noise Awareness Day Lesson Plan: Students use a computing device with a microphone (input) to measure sound levels in their environment. The device then processes this data and displays a result (output) on a website. This lesson provides a clear and practical example of the fundamental input-process-output model of a computing system.

2. Networks and the Internet

As defined in the California K-12 Computer Science Standards, this concept explores how computers are connected to share information and resources.

- Parachutes Lesson Plan: Students create a parachute that will deliver emergency supplies, landing with the lowest impact possible. To monitor impact data, they create a network between an acceleration sensor on the parachute to a central computer able to capture data from each parachute, thus modeling the role of protocols in transmitting data across networks and the Internet.
- Fireflies Lesson Plan: Students learn about the meaning and uniqueness behind the flash of different firefly species. They then design a way to mimic a firefly species using LED lights and Bluetooth connectivity. This lesson mimics actual research done by scientists to learn more about fireflies.
- Infection Game Lesson Plan: A networking simulation where a biological virus spreads through a population, the classroom, via the proximity of one another (via a microcontroller's signal strength). This game demonstrates how information propagates through a network, illustrating concepts like nodes, connectivity, and network topology.

3. Algorithms and Programming

As defined in the California K-12 Computer Science Standards, this concept involves creating and implementing the step-by-step instructions that computers follow.

- Coding Challenges Lesson Plan: Students engage in hands-on programming activities, writing code to solve specific problems. This directly teaches them how to create, test, and debug algorithms in a programming environment.
- Water Turbidity Lesson Plan: In this lesson, students use their skills gained from the Coding Challenges Lesson Plan to develop a turbidity sensor to test water clarity. They will invent a sensor to take readings, connecting the abstract concept of an algorithm to a real-world scientific investigation.

- Atmospheric Pressure Lesson Plan: Similar to the turbidity lesson, students follow an algorithmic process to collect and interpret data about atmospheric pressure, reinforcing the idea that algorithms are essential for both computers and scientific inquiry. They may also use the knowledge gained in the Seeing Color Lesson Plan to create a classroom data display of the current atmospheric pressure.

4. Data and Analysis

As defined in the California K-12 Computer Science Standards, this concept focuses on how computers collect, store, represent, and analyze data to gain insights.

- Earthquake Safety Lesson Plan: This hands-on lesson teaches students about earthquake safety and data analysis. Students build model cities and use a shake table to simulate an earthquake. A micro:bit sensor gathers acceleration data from the model buildings, which students then analyze to find a correlation between building height and shaking frequency. Finally, they use their findings to design safer structures, learning how data can be used to mitigate natural hazards.
- Seeing Color Lesson Plan: This lesson begins with students exploring how humans see the color of pigments and the colors of light. Students then delve into how computers represent colors with ratios of red, green, and blue (RGB) light by learning to program RGB LEDs. Students learn how to make data visualizations using a strip of RGB LEDs as an output to an environmental sensor functioning as the input.
- Secret Handshake Machine Learning Lesson Plan: In this activity, students invent a secret handshake. Then, using a micro:bit Create AI application, they apply a machine learning program to train the micro:bit to detect their handshake as an input. Students may use their knowledge gained in the Seeing Color lesson to incorporate LED light displays as an output. It powerfully illustrates how students may program a computer to use data to recognize patterns and react with a specific output.

CSI facilitator demonstrating the Earthquake Safety hands-on PL lesson.

“I plan on implementing the earthquake lab this year.”
-CSI participant

5. Impacts of Computing

As defined in the California K-12 Computer Science Standards, this concept explores the social, ethical, and cultural impacts of computing technologies.

- High-Low Lesson Plan: This lesson examines the real-world impact of a technological solution to a community problem. Students evaluate the Wello Waterwheel, an invention designed to help transport water more easily. They use the "Three Reads Routine" to explore a video about the invention, identifying quantities and asking questions about its effects. The core of the lesson has students calculate the potential impact on the local water supply (the hydrologic system) if the entire village were to adopt this new technology. By designing a system to monitor the environmental effects of the Waterwheel, students directly engage with how technological innovations have societal and environmental consequences, a key aspect of understanding the impacts of computing.

The CSI model lessons provide a robust and thoughtfully structured pathway for students to engage with the five core concepts of computer science. By grounding each activity in the 5E instructional model and aligning it with state standards, the curriculum ensures a consistent, inquiry-based learning experience that builds from one concept to the next. The ultimate success of these lessons hinges not just on their design, but on their execution with students. The following section will delve into the practical application of this curriculum.

CSI educators at CSPDWeek Anaheim

"It has greatly impacted my project on national parks. I was able to tune it up and flesh it out."

-CSI participant

Implementation & Fidelity

The development of the CSI model lessons was an iterative process, drawing from multiple sources. Some model lessons were adapted from existing science, math, and art materials developed prior to the grant, while others originated from the curriculum of career-focused summer camps. These early experiences helped confirm that the pedagogical approach - using a 5E lesson format with physical computing to spark interest in STEM-related careers - was effective at inspiring student interest in pursuing CS further.

Furthermore, the CSI model lessons were designed to complement the CS curriculum to which participating educators had access for the duration of the CS4NorCal project. To participate in CS4NorCal, middle schools committed to delivering 50 hours of CS standards-based instruction, while high schools committed to delivering 75 hours. The model lessons were flexible; in some cases, they were used to supplement standalone CS curricula like Exploring Computer Science or Code.org's CS Discoveries, while in others they formed the core of the integrated CS instruction. Over the course of the grant period, 57 schools and 78 teachers met this implementation threshold.

To monitor fidelity, teachers responded to an annual implementation survey that provided information on how CS instruction was being delivered in their classrooms. Feedback from these surveys and professional learning sessions highlighted several common challenges. Many of these were related to hardware, such as managing and troubleshooting a class set of microcontrollers. Other challenges stemmed from the virtual delivery of professional learning and coaching, which made it difficult for educators to translate hands-on activities to their own physical classrooms.

The design of the CS Integration curriculum was iterative throughout the grant period, largely in response to the needs of the project. The initial suite of model lessons was continuously expanded to address a greater portion of the California K-12 Computer Science Standards. This expansion also served to meet the demand of a corresponding professional learning model that grew more intensive over time, requiring a larger library of high-quality lessons to serve as exemplars for educators. The impact of these lessons on students is detailed in the following section on Student Outcomes.

> "The workshop provided many hands-on opportunities to engage in full computer science lessons. Where materials were needed, the workshop provided the materials I required. I have at least 5 lessons that I can implement this school year with limited additional planning. I had 3 presenters. 2 out of 3 were engaging, warm, patient, and helpful. They welcomed questions and helped me feel included in a new computer science community."
> -CSI participant

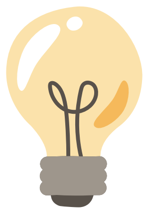
Student Outcomes

To further inform the development and potential impact of the CSI model lessons, a sample of the curriculum was piloted with two classes serving 48 students in total. The feedback from this pilot was very positive, with most students enjoying the lessons and many reporting an increased interest in CS and a greater openness to pursuing a STEM-related career.

The student feedback revealed that the lessons were highly engaging, with students frequently citing the hands-on nature of the activities. On a scale of 1 to 7, the average satisfaction level was approximately 5.2, with many mentioning the fun they had learning to code and seeing immediate results of their work. As one student put it, "It was fun to learn a new thing, and it seems very interesting. I would love to do it again." This engagement translated into a notable impact on students' interest in technology fields. When asked if they were more interested in CS after the lessons, the average rating was 4.6 out of 7, while openness to pursuing a STEM career received an average rating of 4.0 out of 7. In terms of what they learned, students often mentioned the basics of coding and the process of problem-solving. A vast majority stated they would recommend the lessons to others, calling them "fun" and "a good learning experience". The feedback also included constructive suggestions; a few students with prior CS experience found the introductory lessons too easy and suggested starting at a more advanced level.

These positive, small-scale findings align with the results of a large-scale impact study of the broader CS4NorCal grant conducted by UC Davis. The study evaluated the effect of the project's computer science instruction on student achievement in mathematics and science for grades 4 through 8. While the program demonstrated a positive but not statistically significant effect on mathematics outcomes, the study found that the impact on science achievement was both statistically and substantively significant, with students in the program scoring notably higher than a control group. Ultimately, these findings support the core hypothesis of the CS4NorCal logic model: CS standards-based instruction can positively influence student achievement in rural settings, especially when it is integrated into core academic subjects and backed by sustained professional development for teachers.

"I feel confident that I can use the lessons we learned in this class when I get back to school."
-CSI participant



"Again, I loved learning lessons I could implement tomorrow in my class. We learned through struggle."
-CSI participant

Part III - PL Component for Educators

Product Design & Development

The CS Integration professional learning component played a key role in the intensive, two-year sequence designed for CS4NorCal secondary educators. This model was structured to first build a strong foundation in computer science and then focus on classroom application. The first year featured 66 hours of professional learning centered on core CS concepts and curricula. The second year shifted to pedagogy and practical implementation, with 54 hours dedicated to CS integration and application. The CS Integration workshop was originally designed to complement the other curriculum-based workshops offered in this second year, addressing the specific needs of math and science teachers who sought strategies and resources to integrate CS concepts into the courses they already taught.

Based on feedback from participating educators, the CSI professional learning component evolved significantly over the grant period. It was first offered in 2021-22 as a two-day (six-hour) experience embedded within a larger, 66-hour curriculum-focused professional learning experience. By 2024, the component expanded into its own intensive, multi-day, ongoing professional learning experience.

Professional Learning Model (Secondary Teachers)

	Summer	Academic Year
Year 1	30 hours CS Course Specific PL Options: CS Discoveries, Exploring CS, and CS Principles	24 Hours (Just-in-Time) Workshops 12 Hours CS + Core integrated content
Year 2	30 Hours of CS PL Additional Options: Bootstrap Algebra, Equity Minded Instruction, and CS Integration	24 Hours Implement 4 Impact

To ensure teachers felt confident and prepared, the core professional learning was enhanced with hands-on support. During workshop sessions, facilitators modeled lessons, demonstrating effective pedagogical methods for guiding students through the 5E learning cycle and for analyzing and writing code. This collaborative approach included troubleshooting technical hurdles to ensure technology was an enhancement to the learning experience. In addition to this direct support, ad hoc coaching was provided to educators as needed, further supporting them as they implemented the new strategies and curriculum in their classrooms.

Part III - PL Component for Educators

Implementation & Fidelity

The implementation of the CS Integration professional learning component was directly aligned CS4NorCal Logic Model (Appendix A). The primary goal was to enhance teacher confidence, CS knowledge, and pedagogical skills, thereby increasing their capacity to teach to the California K-12 CS standards. This focus on teacher development was intended to lead to improved school and student outcomes, including the expansion of CS pathway options, increased student competency, and greater student achievement in math and science.

While the CS Integration workshop was delivered virtually on multiple occasions to accommodate the geographic distribution of rural educators, the program's designer recommends an in-person implementation. A face-to-face format is considered optimal for facilitating the hands-on, collaborative activities that are central to the workshop's design and for building a strong professional community among participants.

To monitor the fidelity of the professional learning, the project tracked key participation metrics for the CS Integration workshop, consistent with the approach used for all CS4NorCal professional learning activities. Measuring fidelity, particularly through participant attendance and retention, provided a clear measure of whether participants were receiving the intended dosage of the training and informed an iterative redesign of the course. The primary metric was the rate at which participants met an 80% attendance threshold for all workshop sessions. During the 2024 CSI Summer and I4I AYWs, there were 21 participants from across all of California. While 5 of these participants were from the CS4NorCal counties, only 3 of them (60%) met the 80% attendance threshold of 43.2 hours. This percentage is higher than that of the non-CS4NorCal participants, of whom 7 out of 16 (43.75%) attended at least 80% of the AYWs. This recruitment and attrition data were important for evaluating the implementation model and its evolution over the grant period.

To support future replication, the model was designed with flexibility in mind, allowing for a variety of implementation iterations. Because the CSI professional learning component is resource-intensive, a "Professional Learning Toolkit" was created. This document, which is linked in Appendix C, served as a source of truth for participating educators, providing easy access to all model lessons, web-based applications, and other materials used during the professional learning experience. The results of this professional learning are detailed in the following section on Educator Outcomes.

CS Integration facilitator demonstrating building a block coding environment to educators

Educator Outcomes

The CSI professional learning component of the CS4NorCal project resulted in a measurable increase in teacher confidence, as revealed by pre- and post-surveys of 20 respondents. The number of teachers who felt confident enough to teach a standalone CS course grew from 12 (60%) to 15 (75%). More dramatically, the number of educators who felt confident incorporating CS content into another subject, such as math or science, jumped from 13 (65%) to 18 (90%). These results validate the program's core strategies of using accessible, hands-on tools to foster a positive perception of CS through physical computing.

The professional learning sessions were marked by high levels of educator engagement, with participants responding enthusiastically to the hands-on activities using the micro:bit microcontroller. Direct feedback from participants provided further insight into these successes and challenges. Teachers reported that the scaffolded lessons effectively transitioned them from simple code to original creation and praised the variety of presenters, community-building activities, and the immediate classroom applicability of the materials. Conversely, some participants noted challenges related to the pace and volume of content, expressing a desire for more time to delve deeper into building code from scratch.

This feedback was crucial to the program's ongoing evolution. The success of the immersive summer week model, for example, led to the discontinuation of an earlier two-day virtual workshop format. The desire for deeper learning prompted the refinement of content and a partnership with a teacher practitioner to pilot and enhance advanced activities, demonstrating CS4NorCal's commitment to being responsive to educator needs.

The positive outcomes of the CS Integration professional learning are consistent with the findings of the final evaluation report for the larger CS4NorCal project, which engaged 339 educators across 113 schools and educational entities in high-need rural areas. This cohort included 98 standard schools as well as 15 other entities, such as continuation high schools, preschools, and non-profit partners. Of those educators, 166 responded to at least one implementation survey, and 86 responded to more than one survey. Their responses, which likely contain duplicated student counts over time, suggest the project reached an estimated 10,800 students. Motivated by the goal of enhancing college and career readiness, the pedagogical approach of integrating computer science into core subjects using hands-on, inquiry-based methods aligned with broader positive academic outcomes observed in the evaluation. The final evaluation report for the CS4NorCal research project found that computer science instruction had a statistically significant positive effect on student science achievement and a positive, though not statistically significant, effect on mathematics achievement.

Part IV - Next Steps and Closing

Discussion

Learnings and findings from the CS4NorCal project suggest that integrating computer science into core subjects is a powerful and practical strategy for expanding equitable access in high-need rural districts. The positive educator outcomes, such as a significant increase in teacher confidence, paired with the statistically significant improvement in student science achievement, indicate that the CS Integration (CSI) model's design was effective.

A key factor in the project's success appears to be its emphasis on making abstract CS concepts tangible and accessible. By grounding the professional learning and model lessons in physical computing with tools like the micro:bit, the program lowered the barrier to entry for both students and teachers. This hands-on approach likely contributed to the high levels of educator engagement and the measurable boost in their confidence to teach CS. Furthermore, the use of the 5E instructional model provided a pedagogical framework that mirrors the scientific process, likely contributing to the observed gains in student science achievement.

The evolution of the CSI professional learning component also underscores the importance of iterative, responsive design. The program's expansion from an embedded two-day workshop to an intensive, multi-day experience was a direct result of educator feedback and a desire for deeper learning. This demonstrates that for professional learning to be effective in rural contexts, it must be adaptable and continually refined to meet the needs of its participants.

Ultimately, the CSI initiative serves as more than just a curriculum or training program; it is a promising framework for building local capacity. The findings suggest a path forward for various stakeholders. For school leaders, this model provides a strategy to create equitable CS pathways by embedding integrated lessons across core subjects, which can build student interest for more advanced coursework. For curriculum developers, the project highlights the value of designing lessons that use CS as a tool to explore real-world, local phenomena, directly answering the student question of "when are we going to use this?" Finally, for professional development providers, the success of the hands-on workshops demonstrates the impact of using physical computing to build teacher confidence.

“I liked the structure of the lesson presented. I will incorporate the questioning techniques, add asking the students to ask about fellow students' feelings in addition to notice and wonder. I like how this aspect gets the students more involved.”
-CSI participant

“The most helpful aspects were the wonderful collaboration of diverse topics and activities addressing various computer science topics. I learned so much that I can take back to my students. The presenters were patient and creative.”
-CSI participant

Limitations

While the CS Integration model lessons and professional learning component yielded positive results, it is important to acknowledge the limitations that frame the findings. These considerations are essential for interpreting the outcomes and understanding the complexities of implementing educational initiatives in real-world settings.

First, the outcome data, while encouraging, are limited. The results were measured primarily through short-term, post-training surveys of teacher confidence, and there was no long-term follow-up to measure sustained changes in classroom implementation or direct impacts on student learning. As such, the positive findings should be interpreted as measures of participant satisfaction and perceived growth rather than as rigorous evidence of long-term impact.

Second, the generalizability of the findings is constrained by context-specific factors. The program engaged a relatively small, self-selected group of teachers, many of whom may have already been motivated to integrate computer science, which limits the representativeness of the results. Furthermore, the program's design and implementation were tailored to the specific needs of small, rural districts in Northern California and may not translate directly to urban, suburban, or more highly resourced contexts.

Finally, implementation was subject to significant contextual challenges. Early iterations of the program were impacted by the COVID-19 pandemic and devastating seasonal wildfires. The shift to virtual delivery formats for some workshops presented challenges for teaching hands-on physical computing, potentially leading to inconsistencies in participant experience and skill development. Furthermore, teacher and administrator turnover was a constant challenge that affected the continuity of the professional learning cohorts. A teacher's ability to successfully integrate the lessons also depended on factors outside the program's control, such as local access to materials, ongoing administrative support, and available instructional time - resources that are not equally available in all settings.

"They did everything great. I can't think of one area that needs improvement. Perhaps the teachers should be made to sit in random groups each day. This way, with a short introduction time each day, there would be more mixing within the cohort. This would be an uncomfortable part for me, but here I have learned to embrace the adventure and be comfortable while stepping out of my comfort zone."
-CSI participant

Part IV - Next Steps and Closing

Future Research

The current CSI model lessons and professional learning component open several promising pathways for future research that could build upon its findings and address the aforementioned limitations.

First, a more rigorous, quasi-experimental study could be designed to isolate the specific impact of the CS Integration model on student learning. Such a study could compare classrooms where teachers have implemented the CSI lessons against a control group, using pre- and post-assessments of student content knowledge in both science and math, as well as computational thinking skills. This would help establish a clearer causal link between this pedagogical approach and the positive student achievement gains observed in the broader project evaluation.

Second, a longitudinal study could be conducted to investigate the long-term effects of the CSI model on students' academic and career trajectories. This research could track students who experienced the integrated lessons through their high school and postsecondary careers to determine whether exposure to this model correlates with an increased likelihood of pursuing advanced CS coursework or entering STEM career fields.

Additionally, future research could explore the adaptability and expansion of the CSI model lessons and corresponding professional learning. This could involve studies on tailoring the hands-on, inquiry-based approach for other core subjects, such as social studies, or for specific Career Technical Education (CTE) pathways. Research could also test the model's generalizability by adapting the content for urban and suburban contexts to see how it performs outside of the initial rural setting.

Finally, further investigation into the optimal delivery format for professional learning is warranted. A comparative study could explore the differential impacts of in-person, virtual, and hybrid delivery formats. Such research could examine outcomes like teacher confidence, implementation fidelity in the classroom, and the development of a professional community among participants.

"I loved that so much material was covered but it was hard to keep up sometimes. But I have resources to look back on so I did appreciate the volume...a positive and a negative I guess."
-CSI participant

"I really appreciated participating in the lesson as a student to get an idea of the student experience."
-CSI participant

Part IV - Next Steps and Closing

Conclusion

Addressing the persistent gap in computer science education for students in California's rural communities requires a thoughtful, context-responsive approach. The CS4NorCal project was founded on the evidence-based theory that a regional capacity-building model could create sustainable CS pathways for these underserved students. The CSI model lessons and professional learning component were key strategies in this effort, designed to move beyond one-off trainings and provide deep, ongoing support for educators navigating the unique challenges of their small, remote districts.

CSI demonstrated that a thoughtfully designed, hands-on approach to professional development can empower educators to meaningfully integrate computer science into core academic subjects. By grounding instruction in real-world phenomena and aligning instruction with the needs of STEM-related careers, and by leveraging accessible tools like the micro:bit and visual programming, the program not only increased teacher confidence but also sparked student engagement and curiosity. Furthermore, the iterative development of CSI - shaped by educator feedback and logistical realities - underscores the importance of adaptability in educational innovation.

The lessons learned from CSI also offer a path forward. The discussion and limitations presented in this report highlight the complexities of this work, yet the project's overall success provides a compelling case for these pedagogical and professional learning models. The connection between this approach and the statistically significant improvement in student science achievement across the project offers strong evidence for its value. Ultimately, CSI holds promise as a replicable product for expanding equitable access to CS education, empowering educators, and preparing students in rural communities for success in a technology-driven world.

CS educator team during Day 2 of workshops in Redding, CA.

References

Code.org, Computer Science Teachers Association, & ECEP Alliance. (2020). 2020 State of Computer Science Education: Illuminating disparities [PDF]. Retrieved from https://code.org/assets/advocacy/stateofcs/2020_state_of_cs.pdf

Computer Science Education https://www.cde.ca.gov/be/st/ss/computerscicontentstds.asp

Computer Science Teachers Association. (2021). Is computer science education an antidote to rural brain drain? https://csteachers.org/news/is-computer-science-education-an-antidote-to-rural-brain-drain/

Discovery Research. (2016). BBC micro:bit, the next gen: A summary of year 1 of the BBC's "Make it Digital" project. Micro:bit Educational Foundation. https://microbit.org/data/resources/BBC-microbit-The-next-gen.pdf

Gibson, S., & Bradley, P. (2017). A study of Northern Ireland Key Stage 2 pupils' perceptions of using the BBC Micro:bit in STEM education. The STEP Journal: Student Teacher Perspectives, 4(1), 15–41. https://ojs.cumbria.ac.uk/index.php/step/article/view/361

Google & Gallup. (2020). Moving forward: Closing the computer science learning gap—Rural and small-town school districts [PDF]. Retrieved from https://services.google.com/fh/files/misc/closing-computer-science-learning-gaps-rural-and-small-town-school-districts.pdf

Holland, J. L. (1997). Making vocational choices: A theory of vocational personalities and work environments (3rd ed.). Psychological Assessment Resources.

Jones, C. (2019). The long road to college from California's small towns. EdSource. https://edsource.org/2019/the-long-road-to-college-from-californias-small-towns/621428

Lopez, J. M. S., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two-year case study using “Scratch” in five schools. Computers & Education, 97, 129–147. https://doi.org/10.1016/j.compedu.2016.03.003

Smith, S. (2012). Real-world, work-based learning. Journal of Vocational Education & Training, 64(3), 247–260. https://doi.org/10.1080/13636820.2012.688581

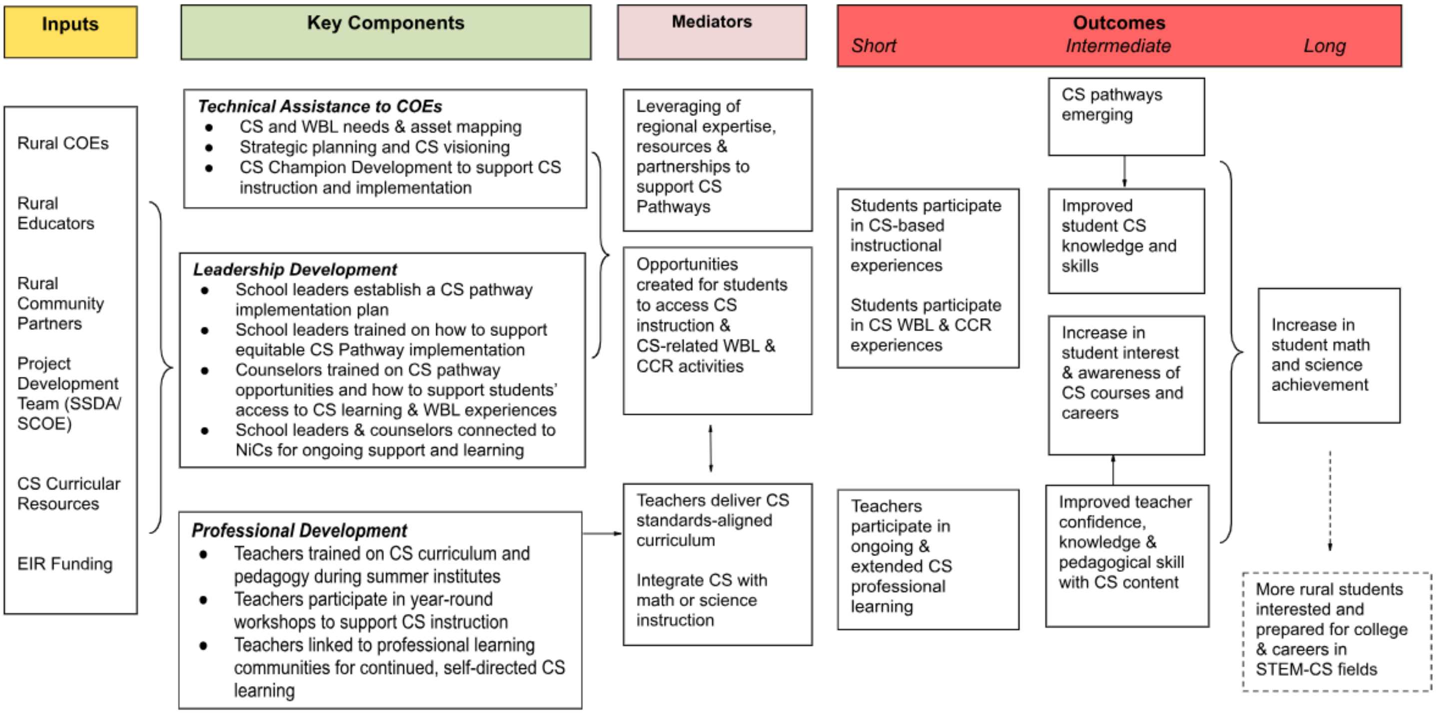
UCLA Center for the Transformation of Schools. (2021). Teacher education deserts in California's rural communities: The role of local, high-quality, and affordable pathways into the profession. https://transformschools.ucla.edu/research/teacher-education-deserts/

Yen, J. M. F., Johansen, A. M., Henderson, T. C., Marx, S., & Grout, B. (2018). Teachers as co-designers of integrated math and science computational thinking curriculum. 2018 ASEE Annual Conference & Exposition. https://doi.org/10.18260/1-2--30991

Appendices

Appendix A - CS4NorCal Logic Model

Program: Rural STEM-CS Pathways Implementation Logic Model
Problem: Lack of access to STEM-CS-based instruction and opportunities in high-need rural areas inhibit potential college and career opportunities for students in high demand industry sectors.
Theory of action: A regional capacity-building approach to creating and supporting STEM-CS pathways that is adaptive to local context can promote an innovative field-based approach to providing students in high-need rural areas with a progressive continuum of exposure to STEM-CS instruction and experiences.



Appendix B - Lesson Plans

Table 2. CS Integration: Implementing for Impact Course Daily Agenda

Workshop	Title	Work-Based Learning Connection
Summer Day 1	Hi-low lesson	Career Awareness: Multiple STEM Fields
Summer Day 2	Earthquake Safety	Career Awareness: Seismologists, architects, civil engineers
Summer Day 2	Parachute Delivery	Career Awareness: Flotation Systems Engineering
Summer Day 3	Investigating Water Quality	Career Awareness: Hydrologists and Water Engineers
Summer Day 3	Simulating Virus Spread	Career Awareness: Epidemiologists and Microbiologists
Summer Day 4	Seeing Color	Career Awareness: Optometrist and Graphic Artist
Summer Day 4	E-Textiles	Career Awareness: Public Servants, Clothing Designers, and Fabric Technologists
Summer Day 5	Noise Awareness	Career Awareness: Audiologists, Sound Engineers, Neuroscience, and Biophysics
Annual Yearly Workshop #1	Project-Based Learning	Career Awareness: Local problem solving
Annual Yearly Workshop #2	Agile Project Management	A project management methodology used in the local industry
Annual Yearly Workshop #3	Emerging Concepts in CS	Career Awareness: Cyber Security, Artificial Intelligence, Educational Technology
Annual Yearly Workshop #4	Showcasing Student Work	Career Awareness: Showcasing solutions to the local industry

Appendix C - Professional Learning Toolkit

CS INTEGRATION
FOR SCIENCE AND MATH

Welcome to CS Integration for
Science and Math Educators Course!