

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/309241593>

Student Centered Observation Protocol for computer-science Education (SCOPE) Version 2.0 -- Code book

Research · October 2016

DOI: 10.13140/RG.2.2.27113.90724

CITATIONS

2

READS

66

8 authors, including:



David Webb

University of Colorado Boulder

35 PUBLICATIONS 434 CITATIONS

SEE PROFILE



Hilarie Nickerson

University of Colorado Boulder

12 PUBLICATIONS 76 CITATIONS

SEE PROFILE



Ryan Grover

University of Colorado Boulder

7 PUBLICATIONS 33 CITATIONS

SEE PROFILE



Kris D. Gutierrez

University of California, Berkeley

95 PUBLICATIONS 3,527 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Scalable Game Design [View project](#)



Social Design Experiments [View project](#)

Student Centered Observation Protocol for computer-science Education (SCOPE)

Version 2.0

David C. Webb, Susan B. Miller, Hilarie Nickerson, Ryan Grover & Kris Gutierrez

University of Colorado Boulder

Webb, D.C., Miller, S.B., Nickerson, H., Grover, R., & Gutiérrez, K. (2014). Student Centered Observation Protocol for computer-science Education (SCOPE). Boulder, CO: University of Colorado Boulder.

This work is supported by the National Science Foundation under Grant No. DRL-1312129 and CNS-1138526. 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 the National Science Foundation.

Purpose

Based on data collected from a study of teachers' implementation of programming lessons in middle schools, we observed an emerging relationship between teachers' classroom practices and patterns in student motivation data. We theorized that pedagogical approaches influence mediation and other instructional decisions employed by teachers as they implemented these units, which impacts student performance, ownership, engagement and efforts to broaden participation (in particular, with females in computer science education) (Webb, 2012). This observed relationship, however, is preliminary and requires further systematic study involving the development and validation of a classroom observation protocol focused on teacher-student and student-student interaction and mediation.

Background

The Observation Protocol for computer science Education (SCOPE) was based on the Observation Protocol for Social Organization of Classroom Instruction (OPSOCI), which similarly was used in previous research studies to document teachers' pedagogical practices and is based heavily on the norms and content of classroom discourse (Gutiérrez, Berlin, Crosland & Razfar, 1999). OPSOCI was significantly modified through multiple iterations to address additional features that were specific to computer science classrooms. The constructs included with the revised protocol are designed to capture different observable aspects of pedagogy ranging from teacher-centered classrooms to student-centered classrooms, specifically considering the following pedagogy styles: teacher directed, whole class guided discovery, community based guided discovery, and self-directed discovery.

Further details regarding the development, internal and external validation, and application of this instrument will be included in future versions of this codebook, protocol and related publications.

References

- Gutiérrez, K., Berlin, D., Crosland, K., & Razfar, A. (1999). *Social organization of learning classroom observation protocol*. Los Angeles, CA: Center for the Study of Urban Literacies, Graduate School of Education and Information Studies, University of California, Los Angeles.
- Webb, D. C., Repenning, A., & Koh, K. H. (2012). Toward an Emergent Theory of Broadening Participation in Computer Science Education. *Proceedings of the 43rd ACM technical symposium on Computer Science Education - SIGCSE '12* (pp. 173 – 178). Raleigh, NC.

Page 1, Section 1 – Class Information.

This section is designed to gather descriptive information about the class and the teacher. This information should be obtained **during** the observation.

Title of item	Description/Explanation
Observer	Enter the name of the observer. If there is more than one, enter both names. If the observers were observing different portions of the class, each observer may choose to complete his/her own observation form.
School	Enter the complete name of the school – do not abbreviate
Teacher	Enter the teacher’s first and last name
Course title	Enter the course title as described by the teacher
Date	Enter the date of the observation
Time start:	Enter the time the observation began
Time end:	Enter the time the observation ended
Grade level:	Enter the grade(s) of all students in the class at the time of observation.
Teacher Gender	Enter the gender of the teacher
Teacher Race	Enter the race of the teacher as described by the teacher.
Course Type	Determine from the teacher if the class is a required course or an elective course. Select the appropriate button.
How many students per computer	Look at the class and determine how the class is set up. Use the MAJORITY of students to make this determination. That is, if MOST students are one student per computer, but two have to double up due to resources, list the class as One Student per Computer.
# Male	Enter the number of male students present in today’s class. (Use actual numbers present, not the class roster.)
# Female	Enter the number of female students present in today’s class. (Use actual numbers present, not the class roster.)
Describe the demographics/Language of the class.	Explain the demographics as specifically as you are able. For example, the class is mostly white, with two minority students (one African American and one Latino). List the language spoken in class, both by the teacher and by the students.
Describe today’s lesson: (For example, third day into Frogger Unit)	Describe the class in detail. What are the students working on? How many days have they been working on this project? If they are all working on different projects, explain what the class has been doing in previous days to lead up to this. Input from the teacher will be essential to answer this question.

Page 1, Section 2: Description of Instructional Style

This section is designed to gather information about the instructional style occurring in the classroom. This section should be completed at the end of the observation. For all percents, round to the nearest 10%. Be sure all percents add to 100%.

Title of item	Description/Explanation
Instructional types	Identify the percent of time spent with the following types of instructional methods as described below. Percents should add to 100%.
<i>Whole Class</i>	Percent of time student and/or teacher discourse was directed towards the entire class
<i>Small Group</i>	Percent of time student and/or teacher discourse was directed towards small groups (2-6 students)
<i>Individualized Work</i>	Percent of time student and/or teacher discourse is directed towards individual students, students working on computers on their own, and other individual work
<i>Other</i>	Percent of time in which discourse occurs in a context that cannot be categorized as small group, whole class, or individualized

Title of item	Description/Explanation
What are the students doing?	Identify the percent of time students spend on each category listed below. Percents should add to 100%
<i>Planning a game/simulation</i>	Creation of a physical model of simulation or game: this is often described as “computer science unplugged,” an instructional context in which the teacher and students act out and/or discuss the different actions among elements to be programmed. Students may work on this individually as well. Planning may also include students designing agents on grid paper, or writing out their game scenario. Student activity typically involves “hands off the keyboard”
<i>Working on a game/simulation</i>	Percent of time in which students are working on computers or other technology. This is time that the students spend actively working on programming the game (with or without help from the teacher or other students) Student activity involves “hands on the keyboard”
<i>Debugging/Testing activities</i>	Percent of time in which students are deliberately troubleshooting their game as a separate activity (not as part of normal programming activities). For example, this might be seen toward the end of game design when other students are testing games for their friends.
<i>Other</i>	The percent of time allocated to activities that cannot be described as individual computer work or engaging students in a physical model of the game or simulation (e.g., taking a test,

Student Centered Observation Protocol for computer-science Education (SCOPE)

	fire drill, etc)
Title of item	Description/Explanation
How did the students and teachers interact?	Identify the percent of time spent with the following types of teaching methods as described below. (Adds to 100%)
<i>Teacher directed</i>	Teacher provides directions/lecture/demonstration style to the entire class. Students may be given instructions (written or verbal) and asked to follow them nearly exactly. Teacher directed instruction can be accomplished as one complete episode with the entire class (e.g., the teacher spends the entire period telling students step by step what needs to be done) or by chunking the work into shorter instructional sections (e.g., the teacher provides occasional direction/lecture/demonstration style instruction to the entire class, followed by significant periods of time in which students work on their own to implement results consistent with the teacher instruction). <i>Students using a step-by-step tutorial that highlights the exact code to use would be classified under this category.</i>
<i>Whole class guided discovery</i>	The teacher engages the whole class in discussions of game design or game creation. As a class, students are invited to reason through, with teacher support, successful algorithms or processes for programming. The teacher <u>does not</u> provide step-by-step instructions as the primary means of instruction. Instead, the teacher often prompts students to determine the appropriate steps. Students have opportunities to contribute to decisions regarding design or programming and make sense of the programming structure or logic. Student-teacher interaction during teacher instruction will occur; student-student interaction during individual work is minimal.
<i>Community centered guided discovery</i>	Similar to Whole Class Guided Discovery in that the teacher may engage the whole class for initial discussions, or when multiple students are struggling. The difference between whole class guided discovery, and community centered guided discovery is that the teacher creates frequent opportunities for students to interact with one another to find appropriate solutions to programming problems in small groups, rather than exclusively with the teacher leading the discussions. Student-student interaction may occur during individual student work time and possibly during teacher-led instruction.
<i>Self-directed discovery</i>	Students work primarily independently with little or no formal teacher instruction (either verbal or written supports) at the whole class level. As an example, this may be observed when students work on games of their own design, or try to create new programming code different from those previously taught, <u>with little to no help</u> from the teacher. There may be student-to-student interaction but that interaction is initiated by the student(s) designing the program.

Describe the student/teacher interactions that led to the above analysis.

Describe in detail the class dynamics that were observed, leading you to make the above determinations. Examples of these interactions are particularly helpful.

Page 2, Section 3: Educational Constructs

This section is designed to note particular educational constructs occurring in the classroom. Be as descriptive as possible within your comments.

1. Who initiates questions/prompts?

Reserved for whole class, unplugged, or debrief type of situations that includes some level of content. (May include assessment and group check-ins) Student-initiated questions or prompts include both with and without teacher calling on them. **Do not include response to questions and prompts, unless said response is its own question or prompt.** Select the category that best describes the overall class experience.

1. Teacher initiates all questions and prompts. Students' answers follow strict IRE protocol and do not themselves ask any questions or prompts.
2. Teacher initiates most questions and prompts. Teacher asks most of the questions and prompts, but there is at least one question or prompt initiated by a student. Can be a clarification question.
3. Teacher primarily initiates questions and prompts, but students frequently ask their own.
4. Initiation of questions and prompts is negotiated between teacher and students. Teacher no longer is the primary initiator. Students may ask questions amongst each other or to the teacher; however, teacher still plays a role and asks questions or prompts, often to keep students on track.
5. Complete absence of teacher in initiation of questions and prompts. This may be because students are leading a discussion or because the teacher's role is to only respond to questions and prompts and not initiate them.
6. N/A No whole class discussion (outside of classroom management)

Be sure to include comments supporting your determination of this construct.

2. Teacher uptake of students' ideas. Whole class

Reserved for whole class, unplugged, or debrief type of situations that includes some level of content. How does the teacher respond to students' ideas? These ideas may be from responses to questions or prompts or may be from the students' own initiation. There may be some instances where the teacher is not present in the conversation (e.g., lead by students). If so, score at least a 3, depending how students respond to each others' ideas. Select the category that best describes the overall class experience.

1. Teacher, at the most, only acknowledges the existence of student ideas. Teacher may evaluate the correctness said idea, but only if it does not build towards the next question or prompt.

Student Centered Observation Protocol for computer-science Education (SCOPE)

2. There is at least one instance where the teacher includes a student's idea and uses it. This may be to use a student's own words to explain a concept or to build towards a bigger idea.
3. If present, teacher often includes students' ideas, as described in (2). However, there are no instances of students using other students' ideas, unless teachers acts as intermediary (e.g., student A has an idea; teacher verifies idea; student B uses idea).
4. If present, teacher often includes students' ideas, as described in (2). There is at least one instance of a student using other students' idea, without teacher intervention (e.g., student A has an idea; student B builds on idea).
5. If teacher is present, both teacher and students often include and use others **students'** ideas.
6. N/A Either no class discussion (outside of classroom management) or no student ideas offered during class discussion.

Be sure to include comments supporting your determination of this construct.

3. Generating other topics. Whole class

Reserved for whole class, unplugged, or debrief type of situations that includes some level of content. The manner in which sub-topics are generated in the course of a lesson activity. (eg. Teacher asks if student's frog is moving...student pipes in asking about what about when it moves on the log.) Select the category that best describes the overall class experience.

1. High frequency of teacher-generated sub-topics. Students are directed to engage in interaction only on these topics. Teacher strictly maintains discussion on these topics by sanctioning or ignoring students' attempts to introduce sub-topics.
2. At least one occurrence of teacher's acknowledgement of student-generated sub-topics.
3. Some instances of teacher's acknowledgement of student-generated sub-topics AND at least one occurrence of teacher utilizing said sub-topic for discussion.
4. Some instances of teacher and students negotiating of ongoing sub-topics of discussion.
5. Regular instances of teacher and students co-constructing sub-topics for discussion throughout the course of the lesson activity.
6. N/A No class discussion (outside of classroom management)

Be sure to include comments supporting your determination of this construct.

4. Discourse Pattern

The overall nature of the pattern of discourse used for Teacher/Student interaction. Select the category that best describes the overall class experience.

Student Centered Observation Protocol for computer-science Education (SCOPE)

1. Strict teacher initiation, student response, and teacher evaluation (IRE) discourse pattern¹. The nature of the student response is short (one word or one phrase) and no response elaboration is encouraged.
2. Strict teacher initiation, student response, and teacher evaluation as described in (1); however, there is at least one occurrence of student-generated elaboration for the purpose of clarification.
3. Relaxed IRE discourse sequence with more student responses occurring in between the teacher initiation and evaluation. Student responses are characteristically longer and the teacher allows students to elaborate.
4. Relaxed IRE discourse sequence with more student responses occurring in between the teacher initiation and evaluation AND student responses occasionally build on previous responses (chained) and contribute to the construction of shared knowledge².
5. Discourse structure is predominantly chained utterances and chained events. Teacher and students build on one another's responses in a manner that closer resembles a conversational discourse structure.

Be sure to include comments supporting your determination of this construct.

5. Computer Science Learning Goals

The overall nature of the instruction geared toward learning computing concepts and practices, including programming. This section covers all of the learning goals specific to a typical computer education class, such as learning about hardware, software, and computational thinking skills (e.g., problem-solving, algorithms, abstraction, simulation, data collection).

Example: Does the teacher explain why a particular element is used, or simply prescribes the use of that element.

Focus on knowing how: A teacher states that there should be a condition in the truck movement rule to "see" the highway in front.

Blend of how/why: A teacher notes that if the truck can move regardless of what is in front, a truck might move on top of another truck, so that a truck should only move forward if there is empty highway in front.

Focus on knowing why: A teacher asks the students to ponder and discuss under what conditions the truck should be allowed to move, so that the class develops the solution.

¹ IRE is a teacher-led, three-part sequence that begins with the teacher asking a student a question or introducing a topic for the purpose of finding out whether the student knows an answer. In the IRE pattern, the student answer is evaluated by the teacher, who makes a brief reply such as "Good," or "No, that's not right." Then the interaction ends. This is in contrast to the Initiation/Response/Follow-Up pattern defined below.

<http://www.learner.org/workshops/tfl/glossary.html#I>

² This is a sequence that begins with either the teacher or student asking a question or introducing a topic. After a response is given, the initiator then uses the response to move the conversation forward. This conversation can continue for as long as the participants wish to talk about the subject, and may include contributions from many people in the class. This approach is in contrast to the Initiation/Response/Evaluation pattern defined above.

<http://www.learner.org/workshops/tfl/glossary.html#I>

Student Centered Observation Protocol for computer-science Education (SCOPE)

Select the category that best describes the overall class experience.

1. The only focus of the computer science learning goals is on knowing how. There is no instance where students learn why they are doing something.
2. The main focus is on knowing how. There is at least one time where students are lead to understand why.
3. There is a mixture of knowing how and understanding why. It is difficult to determine which is the focus.
4. The main focus is on understanding why, but there is at least one instance where students are taught how.
5. Students only learn to understand why. There is no instance of learning how, either because the exercise is to tackle only the why or because the students already know how. This can also include students self-teaching how through their use of why (e.g., students learn why sniffing is useful, but students are left to figure out how to program it on their own).

Be sure to include comments supporting your determination of this construct.

6. Other content goals/purpose

This includes any other goals or purposes that are **not** covered in **Computer Science Learning Goals**. For example, some classes may use computer programming as a means to support another content discipline such as art, math, or a foreign language. This section covers the goals related to those outside disciplines. Select the category that best describes the overall class experience.

1. The only focus of the non-computer science learning goals is on knowing how. There is no instance where students learn why they are doing something.
2. The main focus is on knowing how. There is at least one time where students are lead to understand why.
3. There is a mixture of knowing how and understanding why. It is difficult to determine which is the focus.
4. The main focus is on understanding why, but there is at least one instance where students are taught how.
5. Students only learn to understand why. There is no instance of learning how, either because the exercise is to tackle only the why or because the students already know how. This can also include students self-teaching how through their use of why (Keep in mind that this section relates entirely to non-computer science goals).

Be sure to include comments supporting your determination of this construct.

7. Number of Repairs

Repairs constitute work toward correction, either by students or by the teacher, and include troubleshooting. This can be whole class or individual work. With respect to computer science education, this would also include repair of conceptual or programming issues. This does not include the mechanics of the interface or getting the tool to work (e.g. Click this box to open the file, click this box to save your work, etc...). This can be correcting a statement or question by either the teacher or a student. This also includes verbal or observable self-correction. Correction

Student Centered Observation Protocol for computer-science Education (SCOPE)

can be right or wrong, just as original statement can be right or wrong (e.g., Teacher: “the sky is blue.” Student: “Don’t you mean the sky is green?” counts as a repair). Repairs can also be a process of correction, such as troubleshooting a problem with coding. Select the category that best describes the overall class experience.

1. There are NO repairs that take place during the lesson.
2. There is at least one repair, but they are quite rare.
3. Repairs occur several times, but there are still only a handful of total repairs.
4. Repairs occur quite often within the class (>50%)
5. Repairs occur all the time within the class (>80%)

Be sure to include comments supporting your determination of this construct.

8. Source of Repairs

Who does the correction? The original statement or question that is repaired can be from anyone. Repair can be self-correction. Select the category that best describes the overall class experience.

1. Initiations for repair are generated ONLY by the teacher. This can be the teacher self-repairing or repairing students’ utterances.
2. There is at least one instance of student repair, but overall, the total number of student repairs is very low.
3. There are several instances of student repair.
4. Student repairs occur often within the classroom (>50%)
5. Student repairs occur all the time throughout the lesson (>80%)

Be sure to include comments supporting your determination of this construct.

9. From whom do students receive help?

The manner in which assistance is obtained in the course of a lesson activity. Select the category that best describes the overall class experience.

1. All assistance is obtained from the teacher, either because the teacher is the resident expert, or because the classroom norms are such that students must ask for help from the teacher.
2. Most assistance is from the teacher, with at least one instance of students obtaining help from other students.
3. There is a blend of students obtaining help both from the teacher and from other students
4. More help is obtained by students from other students, than from the teacher, yet there are still occasions where help is obtained from the teacher.
5. All assistance is obtained from other students, either because the teacher is not the resident expert, or because classroom norms are such that students regularly work collaboratively to help others in the classroom.

Be sure to include comments supporting your determination of this construct.

10. Patterns of Engagement

This section looks at which students are engaged. Select each category that best describes the overall class experience.

By gender:

1. Girls Engaged
2. Boys Engaged

By race:

3. Minority students engaged
4. Non-minority students

Be sure to include comments supporting your determination of this construct.

11. What resources were available to students

This section looks at what resources were provided to the students to support them in today's lesson. Resources could be items such as step-by-step instructions on today's task, a general assignment sheet, or general information to support instruction, such as a list of code examples.

- Handout/Printed Instructions – instructions provided on paper to students or groups of students. Describe in comments.
- PowerPoint ® type presentation – instructions provided to all students. Describe in comments.
- Online resources – website to support today's lesson. Describe in comments.
- Information/Instructions posted to board/wall – includes both assignment support as well as general computational thinking pattern support such as might be found on a computer science poster. Describe in comments.
- Teacher computer projected – Teacher screen is projected to model the information on student screen.
- None – no instructions were provided to the students
- Other – describe the resource(s) in the box provided

Be sure to include comments supporting your determination of this construct as well as **whether** and **how** these supports were being used (i.e., are they being used for differentiation? Are they available but not used?)

12. Computational Thinking Patterns

Note the occurrences of specific Computational Thinking Patterns that are used in the classroom. Describe their use in the comments for the overall lesson.

- Absorb
- Collaborative Diffusion
- Collision
- Diffusion
- Generate
- Hill Climbing
- Multiple needs
- Perceive/Act Sync
- Polling
- Pull
- Push
- Script
- Seeking
- Transport

Teacher Interview Questions

The purpose of this interview is to document teacher's instructional decisions and reflections on students activity for the observed lesson. It is also important to know the nature of the instructional materials used with the students: step-by-step tutorials, resources, that support some degree of student decision making, etc. Please be respectful of teacher's time. This should be no more than a 10-15 minute interview for the teacher. When possible, please record this interview

Question 1:

What worked well in today's class?

Probe: How do you know when a class goes 'well'?

In this question, we are trying to understand the teacher's perception of how the class went, as well as what they perceive as characteristics of a 'good' class.

Question 2:

What changes might you make next time you teach this class?

In this question, we are trying to learn more about potential changes the teacher would consider the next time they are teaching the same lesson.

Question 3:

What materials did you choose to use in teaching this lesson? Why?

Probe: Did you change any of the materials? Why?

Probe: In what ways do you feel those materials were the best ones to support your students?

This question will enable us to learn more about the materials the teacher chose to use, and why s/he chose to use those materials. We also want to know how and why changes were made to the materials. (Please reassure teachers that changes are acceptable and expected if they appear to be hesitant to share their thoughts.)

Question 4:

What materials did you use leading up to this class?

While they may not be using any materials in today's observed class, it will be helpful to understand how they have scaffolded support up to this point in the lessons. We would be interested in knowing why types of materials were used, and any details they might share about why they chose that progression in materials.

Provide a summary of the teachers' comments in the boxes provided. If the interview is able to be recorded, please send that recording to the research team.