Helping Middle Schoolers Use Cases to Reason: The Case Interpretation Tool

Jakita N. Owensby (jowensby@cc.gatech.edu)
College of Computing, Georgia Institute of Technology, 801 Atlantic Drive, Atlanta, GA 30332 USA

Janet L. Kolodner (jlk@cc.gatech.edu)
College of Computing, Georgia Institute of Technology, 801 Atlantic Drive Atlanta, GA 30332 USA

Abstract

This paper's focus is on how we can enhance the way students in middle school use second-hand expert cases to reason. We examine the issue in the context of Learning By Design™ (Kolodner, et. al, 1998, 2003) classrooms where students learn science in the context of addressing interesting design challenges. We have previously identified the roles cases can play in helping students learn science better, several difficulties students have with interpreting and applying cases, and the roles teachers can play in guiding students as they grapple with those difficulties (Owensby & Kolodner, 2002). We have also developed a software tool, the Case Application Suite (Owensby & Kolodner, 2002) to help students through some of the difficulties. Its design is based on suggestions from the transfer and case-based reasoning literatures and the approach to education called cognitive apprenticeship. In this paper, we discuss the design of the software, present data from an early analysis of its effectiveness, and attempt to draw some preliminary conclusions about the feasibility of dividing up responsibilities for scaffolding student learning between teacher, peers, and software tool.

Introduction

We ask a broad question in our research study: How can we help students as they attempt to understand an expert case, pull lessons from it, and apply those lessons to their designs as they are reading cases to learn science; and if we can help them learn this skill, will it transfer to new situations that require the ability to interpret and apply a second-hand experience? Our software, the Case Application Suite, is designed to provide scaffolding[1], supplementing that of the teacher, to help students not only engage in applying cases, but to also learn the skills necessary to effectively engage in those processes, thereby serving as a model for promoting skill learning. We propose that if we help students as they actively engage in case application, the skills needed to carry out this process will transfer to other areas of their lives. In this paper, we ask if the kinds of scaffolding used in our software work, and we begin to analyze student internalization of the skills and perception of the processes involved in case application. We address these issues in the context of project-based inquiry science, an approach to science education that has students learning science content and scientific reasoning skills in the context of addressing interesting challenges or driving questions. We have identified several affordances that cases have as resources along with some of the difficulties students have with interpreting and applying them and roles the teacher can productively play in guiding students as they grapple with those difficulties (Owensby & Kolodner, 2002). Our software suite, the Case Application Suite, is designed to scaffold students as they apply expert cases to addressing a project challenge. It includes three tools—Case Interpretation, Case Application, and Solution Assessment. We've piloted it with sixth and eighth grade students in Learning by Design (Kolodner et al., 1998, 2003a, 2003b) classes as they learn geology in the context of designing a tunnel. Expert cases presented in the unit are intended to inform their inquiry, designs, and decisions. Students collaboratively discuss interpretations of each case, best ways to apply what they can learn from it, and best ways to assess the potential success of their application ideas, using prompts in the tool suite as reminders about how to do those assessments. This paper seeks to describe the pilot of the software as well as preliminary results from observations, performance assessments, and student interviews, data that we have collected over the course of this pilot, the full analysis of which is still in progress.

Background

Our research group has been developing project-based inquiry science units for middle school based on the literature on how people learn (e.g., Bransford et al., 1999) and the cognitive model suggested by case-based reasoning (Kolodner, 1993, 1997). Inherent in that literature is the importance of repeated practice, giving students numerous opportunities to engage in carrying out a task, get feedback on their performance and help with debugging and refining their performance, thereby, gaining a better understanding of the skills needed for that task. Our approach, called Learning by Design (Kolodner, et al., 1998, 2003a, 2003b) focuses on design challenges. In our physical science units, students design and build a device, testing it and getting feedback on how well it works, explaining its deficiencies, and trying again. For example, to learn about motion and forces, students design and build a miniature vehicle and its propulsion system. For earth science, however, it is

[1] Scaffolding is a term used in the developmental and education literatures to denote help that is given to the learner on an as-needed basis to help the learner achieve some goal or perform some task that he/she could not do alone. The best scaffolding not only helps the learner perform better but also can contribute toward helping the learner develop the capability of performing the task him or herself.

impossible for students to design and build a working artifact. We have had to find a different way for them to get feedback on their design decisions. One way is to compare their design solutions to solutions used by experts in real situations. Such cases can also provide suggestions to students as they are designing.

The Case Application Suite is designed to help students interpret and apply cases and assess their design solutions. Its design derives from two foundations – case-based reasoning (Kolodner, 1993) and cognitive apprenticeship (Collins et al., 1989). Cognitive apprenticeship suggests that skills learning can be promoted if learners have the opportunity to see the skills they are trying to learn modeled in context of use and then have opportunities to be coached through doing them themselves, within a context where those skills are needed and used repeatedly. Case-based reasoning begins articulating the processes involved in using experiences to solve problems, suggesting that interpretation of cases to extract rules of thumb, their applicability, and why they work will enhance the ability to use cases well (Kolodner, 1993, 1997).

Case Application Suite

The Case Application Suite was designed to play the role of coach as individuals or small groups of students are working to interpret an expert case and apply it to their challenge, asking the kinds of questions and making the kinds of suggestions that can push the students' understanding, helping the students make connections and develop the ability to justify their decisions with evidence. Its questions prompt students to reason as CBR suggests, e.g., to identify the rules of thumb that they can glean from a case, analyze their applicability to the new situation, and consider alternatives. Hints and examples are provided with each prompt to give students clues about the kinds of things they can include in their answers to make them informative.

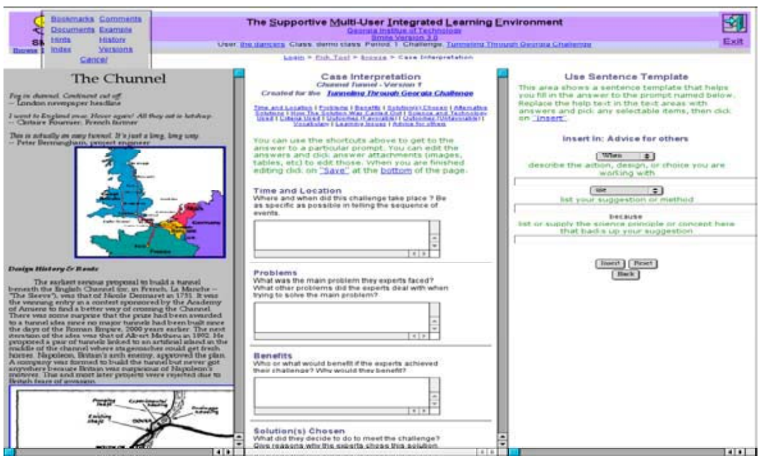

Figure 1: Case Interpretation Tool with the rule of thumb template in the right hand frame and the Browse menu used to access examples

The Case Application Suite contains three tools: Case Interpretation, Case Application, and Solution Assessment (Owensby & Kolodner, 2002). Each has a left frame that displays the case or previous linked artifacts; a middle frame with prompts; and a right frame that displays hints, examples, comments, and templates. Figure 1 shows the Case Interpretation Tool. In its left-most frame is the case being analyzed; in its middle frame are prompts in the form of questions, and in its right frame are hints for articulating a rule of thumb.

Students are scaffolded in four ways. First, the structure of the suite serves as a scaffold. Each tool corresponds to a major step in the case application process and together they provide a high level view of the processes involved in case application. Second, the prompts in each tool's center frame sequence the steps involved in carrying out each high-level process. Third, hints are provided with each prompt to help students carry out each step. Finally, examples are provided to help students understand what they need to be accomplishing. The Case Interpretation Tool helps students identify problems experts encountered in achieving their goals, solutions they attempted and why they chose those, what results they accomplished and explanations of those, and any rules of thumb, or lessons, that can be extracted from the experience. The Case Application Tool guides students through attempting to apply the lessons of a case, prompting them to consider whether a rule of thumb is applicable and how to apply it. The Solution Assessment Tool helps students assess the potential success of their solution, analyzing the impacts they expect their solution to make. Table 1shows prompts from the three tools. Students use the tools as templates, working in small groups to fill in answers to its questions (middle columns). The prompts in that column together with the hints and examples in the right column provide reminders about what needs to be considered to do a complete job of the task.

The tool suite is designed to complement modeling and scaffolding provided by the teacher and as an aid to student sharing their work. When integrated well into a classroom, the teacher models the stages of case use and coaches students, as a class, as they attempt to interpret or apply a case or assess a solution. The software is then used small groups as they carry out those processes without the teacher. Then groups present their cases to the class followed by a teacher-facilitated discussion about what can be learned from the full range of cases and the reasoning the most successful students engaged in to do their work.

Table 1: Case Application Suite Prompts

Case Interpretation Tool	Case Application Tool	Solution Assessment Tool
Time and Location Where and when did this challenge take place? Be as specific as possible in telling the sequence of events.	**Our Design Goals** What are your design goals? List them separately.	**Assess Solution (Design Goals)** Which specific design goals are successfully met by your new solution? Which are not?
Problems What was the main problem they experts faced? What other problems did the experts deal with when trying to solve the main problem?	**Our Problems and Sub-Problems** What problems and sub-problems do you face in your challenge?	**Potential Problems That Were Seen Along The Way** Which specific issues and sub-issues are successfully met by your new solution? Which are not?
Benefits Who or what would benefit if the experts achieved their challenge? Why would they benefit?	**Our Criteria and Constraints** What criteria and constraints are present in your challenge? How do they affect each of your design goals? Issues and sub-issues?	**Assess Solution (Criteria and Constraints)** Were the criteria and constrains in your challenge taken into account by your new solution? How?
Solution(s) Chosen What did they decide to do to meet the challenge? Give reasons why the experts chose this solution.	**Rule(s) Of Thumb** Looking back at the rule(s) of thumb that were created in the Case Interpreting Stage, which ones, if any do you think will help you solve your challenge?	**Things Overlooked** Were any criteria and constraints overlooked? How?
Alternative Solutions Were other ways of meeting the challenge that were considered? Why were they not chosen?	**Problems Addressed** What problem does this rule of thumb address in your challenge?	**Next Steps** If design goals, issues/sub-issues, or criteria/constraints were not met, decide if your current solution covers enough to stand alone, whether it should be meshed with another solution to make a more complete solution, or if it should be abandoned.
How The Solution Was Carried Out How did they put the solution into practice? What steps did they take to carry the solution out ?	**Criteria/Constraints** What criteria/constraints does this rule of thumb satisfy in your challenge? ...	
Science and Technology Used What science and technology were used in choosing the solution? In putting the solution into practice?	**Predictions** What predictions can you make about the outcomes (favorable and unfavorable) of your solution if you apply this rule of thumb? If you don't apply it?	
Criteria Used What criteria were used to select a solution? To select how the solution would be put into practice?	**Is This Rule Of Thumb Helpful** Is this rule of thumb helpful to use to design a solution for your challenge? Justify.	
Outcomes (Favorable) Were any of the outcomes favorable? What short-term effects did these outcomes have? Long-term?		
Outcomes (Unfavorable) Were any of the outcomes unfavorable? What short-term effects did these outcomes have? Long-term?		
Learning Issues Do you know everything you need to know about this case to move on? Do you have any questions about whether this case can help you solve your challenge?		
Advice for others Do you have any 'rules of thumb' for others?		

Our Study

We were interested in finding out how effective the kinds of scaffolding used in our software are when teachers also carry out their parts in a cognitive apprenticeship, especially (i) whether the tool enhances student use of cases when they are using it, (ii) whether use of the tool for some case interpretation enhances their ability to use expert cases to reason when they do not have the tool available, and (iii) whether use of the tool enhances students' understanding of the processes involved in case application.

We piloted the Case Application Suite in the classrooms of two LBD teachers during the Fall 2002 semester, a 6th-grade teacher (Mr. J), and an 8th-grade teacher (Mrs. K) (Owensby & Kolodner, 2003a). The analysis presented here is based on Mrs. K's students. The software was used as part of the LBD *Tunneling* unit. In *Tunneling,* each group, comprised of three or four students, serves as a team hired to help design a tunnel for a transportation system that will run across the state of Georgia. Four tunnels need to be designed, each for a different geological area of the state – one mountainous, another in a sandy region, and so on. They need to address several issues, e.g., at what depth to dig the, what methods to use for the digging, and what support systems are needed in the tunnel infrastructure. Cases are provided as a means of introducing students to different kinds of technology used to build tunnels, to give them an appreciation of the difficulties and dangers they must take into account when designing their tunnels, and as examples of approaches that worked well, that they can apply to their designs, and approaches that did not work so well, so that they can avoid making the same mistakes. Planning to avoid those mistakes usually requires learning more about the geology of the tunnel region.

We analyzed data from students in two of Mrs. K's classrooms. One class had 29 students in 8 groups, the other had 18 students in 5 groups. In the first class, 2 groups (8 students) used the software; in the second 2 groups (6 students) used the software. Students who did not use the software (n = 33) instead used less detailed paper-and-pencil design diary pages (Puntambekar & Kolodner, 1998) to scaffold their work. The My Case Study diary page is a chart with four columns labeled Case Summary, Problems that Arose, How *Problems Were Managed, and Ideas To Apply To Our Challenge.*

The Environment of Use

Mrs. K's implementation of the Tunneling unit was close to the one suggested by LBD materials; she modeled cognitive skills as a Cognitive Apprenticeship approach suggests. For case interpretation, for example, Mrs. K created a template similar to the Case Study design diary page that had columns labeled *Facts, Problems, Solutions, Constraints, Ideas/Rules of Thumb, and Questions,* that she transferred to a large piece of yellow paper hung at the front of the classroom. She read one case (the Lotschberg Case) with the students, and as they identified facts, problems, and solutions, she wrote them on the yellow paper under their respective headings. Afterwards, she helped the class think about and state rules of thumb that could be derived from the case (lessons learned), and she wrote them on the yellow paper. From those rules of thumb, she pushed the students to formulate questions (i.e., things they needed to learn more about in order to apply the case) that she wrote under the Questions heading. At the end of this exercise, she informed the class that they would do the same thing with the cases that they would be assigned, and she assigned a case to each group. Students using the software gathered around the computers; those not using software sat around tables and read their cases together using their design diary page. When they were finished, they could go to the library to search for information that would help them answer their questions. This was followed by class presentations of the cases and discussion of each – both to make clear what each case could teach and to make clearer how to interpret cases well. Her implementation of case interpretation lasted 6 class periods. Then students moved on to applying those cases to their tunneling challenge.

The classroom had a network of four computers, a laptop server, and a printer. The computers were next to each other, so students using them were together in one area of the classroom. If a group had 3 students, they worked around one computer. If it had 4, they worked in groups around two adjacent computers.

Data Sources and Analysis Procedures

We collected data from three sources. (i) We observed students using the software in order to understand how students used the scaffolding and discover any deficiencies with the software and making presentations to their peers of their cases to capture the quality of their analysis and use of the cases they were reading. (ii) We conducted a performance assessment to understand how well, if at all, students could apply acquired skills in a new situation and to see if there were differences in this ability between students who used the software and those who did not. (iii) In structured interviews, we asked students who had used the software questions to uncover their perspectives about the software and what they could express about the processes involved in case application.

In the performance assessment, called the Bald Head Island Challenge, each group played the role of a design team that has been hired by a construction company to investigate the feasibility of building two subdivisions on an island off the coast of Georgia. The company has allotted two million dollars and ten months to investigate, plan, and construct the subdivisions. The structure, layout, and composition of the island off the coast of Georgia are very similar to the Bald Head Island. The performance assessment included four tasks: finding and describing the risk factors involved in building subdivisions on the island, identifying and describing possible ways to manage those risks, designing a plan to test the management methods identified, and creating rules of thumb that can be used in future projects like this. Although the groups discussed their answers together, each group member was responsible for writing up his/her own answers.

To help them, students were given a chart with five columns labeled *Describe the risk, Why is this a risk, What are ways to manage the risk, Pros*, and *Cons*. The chart was similar in structure to My Case Summary pages and the one created by Mrs. K, so it was something all the students were familiar with using. It provided some organization, but only a minimal level of scaffolding. The groups that the students were in during the performance assessment were the same groups that students were in as they engaged in the Tunneling Unit; the breakdown of groups was the same as described earlier. In each period, five groups were videotaped.

We've analyzed student performance for the first two tasks on dimensions including *Recognizes that the case should be used to solve the challenge, Makes direct reference to the case to justify an argument or position, Able to identify expert problems*, *Able to identify expert "mistakes," Able to identify relevant aspects of the case that can be applied to the challenge, Identifies risks based on prior experience with another LBD/software case, Able to identify criteria and constraints, Uses the case to understand the context of the risks, and Identifies rules of thumb*. We used a five-point Likert scale for each, with one representing no evidence of the quality to be rated, and with five representing the quality completely captures the nature of the episode. Three coders, two independent raters and the first author, coded the videotaped performance assessments as well as the accompanying charts for all students (comprising all four tasks)[2]. Each coder coded independently after which all coders discussed variations in rates given and a negotiated rating was assigned.

Results

Observations

Software groups were videotaped as they used the software, and software and non-software groups were videotaped as they presented their initial case studies. In addition, the yellow charts were collected for analysis. Software groups tended to talk across computers, and they always converged on a group answer before entering an answer into the computer. The quality of the presentations in Mrs. K's class was very high among both the software and non-software groups. But examining their artifacts and videotapes of the presentations shows three differences. First, the software groups made connections between being unable to satisfy criteria and constraints and the outcomes that result. Second, they had more sophisticated causality to their rules of thumb. For example, the non-software groups' rules of thumb are in the form of simple imperative statements (e.g., "Control water problem", "Take core samples"), while the software groups' rules of thumb explain why (e.g., "Take core samples—they can save your life because if you hit the wrong kind of rock, you can get hurt", "You should always have an oxygen pass so the toxic fumes can get out"). Third, the software groups tended to include the process the experts used to implement the solution. For example, one software group noted that because the experts couldn't drill through the mountain like they wanted to, they had to build the mountain in an S shape. Their non-software counterpart made no mention of this, even though the fact that an S shape was chosen over other alternatives was an important facet of the solution.

Performance Assessment

Preliminary results show better performance by software groups on four dimensions:.

1. Software groups tended to describe expert problems on a finer-grained level than non-software groups, with software groups averaging 4 for that dimension and non-software groups averaging 3.17[3]. For example, non-software groups identified "sand" as a risk, while software groups identified the "incompatibility of the old sand and the beach and the new sand dug when the channel was deepened" as risk, or expert problem. In the case, there are a number of risks or problems that involve sand, so being able to distinguish between those problems is important.
2. Software groups tended to discuss whether a management method made sense for their challenge, analyzing how the management method would play out in their challenge and questioning each other about the feasibility of a proposed management method. Non-software groups discussed management methods only if they were extremely far-fetched (is this what you meant?). Software groups averaged 3.33 for this dimension, while non-software groups averaged 1.94.
3. Software groups considered the role of criteria and constraints when discussing the pros and cons of a management method, while few of the non-software groups mentioned criteria and constraints at all, with software groups averaging 4.3 and non-software groups averaging 2. For example, nearly every software group considered time and money when discussing the pros and cons of their management method.
4. Software groups used the case not only to identify the problems the experts encountered, but also to understand the context in which those problems arose. They wanted to understand what was happening in the environment that caused the problems to occur or to grow worse. Non-software groups tended to look for keywords that they were familiar with when identifying problems and management methods. For example, while flipping through the case, one non-software student declared, "Oh!! I see erosion here—erosion is a problem." In a similar incident in which one software group member stated that erosion was a problem, another member of that group declared, "but it says here that the problem is the shoreline eroding." This discussion resulted in the software group providing more detail about the erosion problem. Software groups averaged 2.66 for this dimension, while non-software groups averaged 1.38.

Interviews

We conducted individual interviews with the 14 students who used the software. Students who used the software thought it helped them organize their thoughts and keep track of where they were in the bigger task better than reading the case, pulling important facts, and writing them up on their own. (They had done that during a previous

2 For a more detailed discussion of the coding schemes used, see Owensby & Kolodner, 2003b. Inter-coder reliability will be determined at the completion of the full analysis of the performance assessment (all four tasks).

3 We are in the process of computing the standard deviations and significance for these results and will have them in time to include them in the poster presentation.

unit.) Students told us they used the hints a great deal, and they thought that the hints generally provided helpful explanations of what they were being asked to articulate, though some hints were clearer than other (thus, uncovering a deficiency of the software). In addition, students thought the prompts helped them readily identify the important aspects of the case quickly and helped them formulate more thoughtful answers. Most interestingly, perhaps, almost every student identified a process for case application that matched what was in the software: identify the main issues/problems in the problem to be solved; identify the main issues/problems in the case; identify solutions in the case; identify criteria and constraints in the problem to be solved; identify criteria and constraints in the case; match problems between problem to solved and case; match criteria and constraints between problem to be solved and case; decide if the solution from the case can be used directly or if some parts need to be changed in order to use it.

Discussion

While our results are still preliminary and only focus on one part of the Case Application Suite's software, results suggest that tool's means of providing scaffolding is effective for at least some important parts of case interpretation and application, and that the computer can indeed successfully share responsibilities with the teacher in a cognitive apprenticeship style classroom, providing help to small groups as they work without the teacher's availability. We are still collecting and analyzing data about the use of the entire suite of tools and about students' ability to transfer what they learned using the tools to other case interpretation and application situations. We are also in the process of finding out to what extent the tools can be effective when a teacher does not model skills as well as Mrs. K does. While the tools were designed to complement the teacher as coach, the reality of the classroom is that not all teachers are skilled at modeling and coaching. We will use what we find to revise the software tools to make them more effective and to further develop teacher materials in an effort to better support them as they integrate these tools into their classrooms. We still need more research about exactly how to divide up the responsibilities between the teacher, software, and peers. During the coming school year, we will be running a more in-depth analysis addressing the bigger questions of our research study.

Acknowledgements

We would like to thank the McDonnell Foundation, the National Science Foundation, and the National Physical Science Consortium for their support in this research effort.

References

Bransford, J.D., Brown, A.L. Cocking, R.R. (1999). *How People Learn*. 41-66. Washington, D.C. National Academy Press.

Collins, A, J.S. Brown, & S. E. Newman (1989). Cognitive Apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Ed.),*Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*. Hillsdale, NJ: Erlbaum.

Kolodner, J.L. (1993). *Case-Based Reasoning*, Morgan Kaufmann Publishers, San Mateo, CA, 1993.

Kolodner, Janet L. (1997). Educational Implications of Analogy: A View from Case-Based Reasoning. *American Psychologist.*

Kolodner, J. L., Crismond, D., Gray, J., Holbrook, J., Puntambekar, S. (1998). Learning by Design from Theory to Practice. In A. Bruckman, M. Guzdial, J. Kolodner, & A. Ram (eds.), *Proceedings of International Conference of the Learning Sciences 1998*. Atlanta, Georgia.

Kolodner, J.L., Crismond, D., Fasse, B., Gray, J., Holbrook, J., Puntembakar, S. (2003a). Putting a Student-Centered Learning by Design™ Curriculum into Practice: Lessons learned. *Journal of the Learning Sciences*, Vol.12 No.4.

Kolodner, J.L., Gray, J., & Fasse, B. B. (2003b). Promoting Transfer through Case-Based Reasoning: Rituals and Practices in Learning by Design™ Classrooms. *Cognitive Science Quarterly*, Vol. 3.

Owensby, J. & Kolodner, J.L. (2002). Case Application Suite: Promoting Collaborative Case Application in Learning By Design™ Classrooms. *Proceedings of the International Conference on Computer Support for Collaborative Learning*, CSCL-2002, Jan 2002.

Owensby, J. N & Kolodner, J.L. (2003a). Case Interpretation Tool: Collaboratively Coaching Students' *Understanding Of Second-hand Experiences In Learning* By Design Classrooms.

Owensby, J.N & Kolodner, J. L. (2003b). Case Application Suite: A Study of Teacher Use in Learning By Design Classrooms. Paper presented at the American Educational Researchers' Association (AERA) 2003 Conference, Chicago, IL.
http://www.cc.gatech.edu/projects/lbd/pubtopic.html#assess

Puntambekar, S. & Kolodner, J.L. (1998). The Design Diary: A Tool to Support Students in Learning Science by Design. *Proceedings of ICLS 98*. Atlanta, GA,