BeeSign: Designing to Support Mediated Group Inquiry of Complex Science by Early Elementary Students

Joshua A. Danish, Kylie Peppler & David Phelps
Learning Sciences, Indiana University
201 North Rose Ave, Wright 4040
Bloomington, IN 47401 USA
jdanish@indiana.edu

ABSTRACT
All too often, designers assume that complex science and cycles of inquiry are beyond the capabilities of young children (5-8 years old). However, with carefully designed mediators, we argue that such concepts are well within their grasp. In this paper we describe two design iterations of the BeeSign simulation software that was designed to help young children learn about honeybees collect nectar from a complex systems perspective. We summarize findings from two studies that suggest that this design has been successful in teaching and motivating these young children and demonstrates how activity theory can guide design.

Categories and Subject Descriptors
K.3.1 [Computers and Education]: Computer Uses in Education - *Collaborative learning.*

General Terms
Design

Keywords
Science Education, Inquiry, Simulation, Complex Systems, Young Children, Interactive Whiteboards

INTRODUCTION
The developmental psychology literature frequently characterizes young children as not very good at scientific reasoning [7] or at designing science experiments effectively [8]. For these reasons, science education for young children is often limited to rather superficial treatments. This approach is problematic because it may lead to children developing early misconceptions which are difficult to overcome in later grades, and because it may lead children to develop an incorrect view of science as consisting of facts rather than a process [7]. In this paper we report on two design and implementation iterations of the BeeSign simulation software, designed by the first author [2] and informed by activity theory. This work supports young children (5-8 years old) in cycles of scientific inquiry as they learn about how honeybees collect nectar from a complex systems perspective. The BeeSign software can be accessed at http://www.joshuadanish.com/beesign.

Complex Systems
Increasingly, scientists view the world as a collection of complex systems composed of multiple interdependent elements whose interactions lead to emergent properties that are more than simply the sum of their parts [5]. Understanding complex systems concepts such as emergence, interdependence, and feedback loops may support students in a host of domains ranging from the sciences to mathematics, economics, and others [3].

Unfortunately, it is quite difficult for students to learn about complex systems [9]; they often pursue a "centralized" explanation, assuming that some entity organizes and directs an entire system. Similarly, while experts typically think of a complex system in terms of the functions that its various components support, novices tend to focus on the superficial structures and behaviors of the system [4]. In order to truly understand a complex system, it is often necessary to think about it at several levels including the local behaviors of individual elements (e.g., honeybees) as well as at a more aggregate level (e.g., the entire hive) [9].

BeeSign
Despite the challenges that students face in understanding complex systems and sophisticated inquiry, the value of such understanding is undeniable. Therefore, given that students' early understandings play such a crucial role in shaping their later learning, BeeSign was designed and implemented in an effort to provide a developmentally appropriate method of engaging students as young as kindergarten (5-6 years old) with complex systems concepts in the context of honeybees collecting nectar [1, 2]. BeeSign is both a simulation tool and a curriculum that includes several other activities designed to support students in learning about how honeybees collect nectar at the local and aggregate levels [9].

In the current paper, we report on the BeeSign simulation software, which was designed to help students focus on honeybee hives from an aggregate perspective. More specifically, BeeSign was designed to help students learn about how bees dance to communicate the location of viable nectar source, which in turn leads to the hive as a whole collecting nectar in an efficient and adaptive manner.

THEORETICAL FRAMEWORK

The design of BeeSign was guided by Activity Theory [6]. Activity theory highlights the fact that individuals learn through social interaction and that various tools mediate interaction. Tools are both material (i.e., they have a physical presence) and ideal (i.e., we associate beliefs with them). Furthermore, activity theory focuses our design efforts upon the object of activity—the larger goals shared by a group. In the design of BeeSign, the intention was that the object of students' activity would be to understand how honeybees collect nectar. As individuals attempt to attain this object, their interactions are presumed to be mediated by the tools they use, the community in which they are acting, the rules that they follow, and the division of labor that governs the various roles that they play.

Therefore, in designing the BeeSign simulation software, it was necessary to consider not only the software itself, but the activity system in which it was going to be used. BeeSign was designed for projection onto an interactive whiteboard to support collaboration amongst small groups of 5-10 students guided by a teacher to facilitate their efforts. In this context, it became possible for the teacher and the software to mediate the students' interaction with the concepts in such a way that they were capable of engaging in productive inquiry about a complex system from an aggregate perspective—activities that would have been difficult if not impossible without this kind of support.

METHODS

BeeSign has now been implemented in two iterations as part of two different studies. The findings presented here are intended to illustrate features that, across both implementations, appear to make BeeSign successful as a tool for supporting students' collective inquiry. In both cases, video data was collected of students using BeeSign. We analyzed the video in iterative cycles as proposed by Erickson and other Discourse Analysts to develop hypotheses about how the BeeSign interface supported students' activity and learning. These hypotheses were then revised through repeated viewing of the video and discussion amongst the research team.

BeeSign 1.0

The first implementation of BeeSign took place with 42 kindergarten and first-grade students (5-7 years old) in a progressive elementary school in southern California. The results of this study demonstrated that, based on pre- and post- interview results, students' understanding of how honeybees collect nectar increased significantly over the course of the curriculum [1, 2]. Furthermore, it appears that BeeSign was the key component in helping the students to understand how the honeybees collect nectar from an aggregate perspective.

BeeSign 1.3

BeeSign 1.3 was recently implemented as part of the Cross-Curriculum Representational Practices project (CCRP). This project took place with 44 first and second grade (ages 6-9) students in a public elementary school located in central Indiana. While this study was only recently completed, preliminary analyses suggest that students learned about honeybees and the nectar collection process.

In both cases, the BeeSign software was part of a larger curriculum that also included activities such as students drawing and reading about honeybees, engaging in participatory simulations where students act out the role of a honeybee in a game-like setting, and creating participatory models, where students create a skit to demonstrate their understanding of the bee behaviors.

BEESIGN FEATURES AND FINDINGS

The main interface of BeeSign (see Figure 1) consists of two simulation windows. Each window includes a hive and a title display that describes the behavior of the honeybees in the hive in terms of whether or not the bees dance if they find a nectar source (as honeybees do), whether they simply return to the source of nectar (as bumblebees do) or whether they forget entirely (a purely hypothetical scenario for comparison purposes). The two windows may each be assigned a different behavior. Flowers are then placed in each window. When the simulation is started, students can observe simulated bees searching the space around the hive for flowers, and then returning to the hive after they have either found nectar or reached the limit of their flight range.

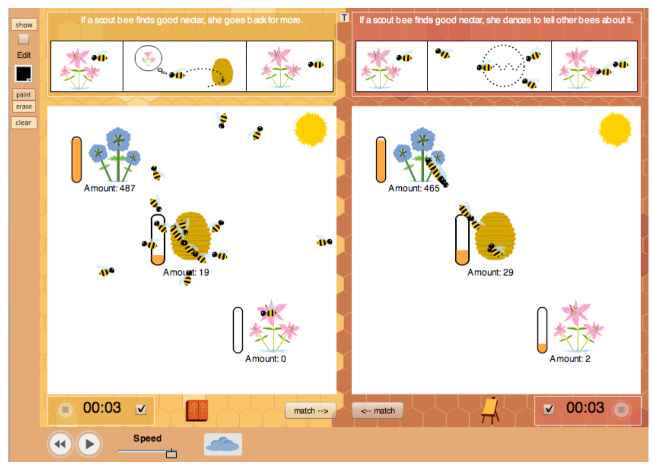

Figure 1: The BeeSign Interface

The teacher may choose to either focus the students' attention upon the bee flight patterns, or the amount of nectar present in the flower. The nectar amount may be displayed using a simple bar reminiscent of video games. le

We will now briefly discuss three aspects of the BeeSign design across both implementations: the way that it supported teacher-led group activity, successfully scaffolded inquiry, and helped students to focus on patterns in the aggregate behavior of the hive.

Supporting Group, Teacher-Led Activity

As noted above, the first author designed BeeSign [1, 2] to be used by groups of students with a teacher leading the discussion. To support this, BeeSign was specifically designed to project on an interactive whiteboard so that students could both see and interact with the software easily (see Figure 2). This design included a number of choices about which features were most likely to be used by the students (controlling the simulation playback, and adding new flowers to the simulation). These features were placed lower on the interface, at the height that the children could easily access them (note the placement of key buttons below the simulation windows in Figure 1). Finally, because it is often difficult to trigger a mouse-over or right-click on an interactive whiteboard, the BeeSign software was designed to work without these interface features.

Figure 2: BeeSign in use

To complement the design of the BeeSign software for supporting group interaction, a typical BeeSign session consisted of asking students to make predictions using the interactive whiteboard and then discuss these predictions to make their thinking visible. Then, the simulation was run and students would describe their observations, often using the interactive whiteboard to illustrate them. A typical cycle might last 10 minutes with 4-5 minutes being devoted to each the prediction and observation with only a minute or so devoted to the actual BeeSign simulation.

In BeeSign 1.0, students drew their observations directly onto the whiteboard where the entire group could then discuss them. While this was effective, it made it difficult at times to see the simulation underneath the drawings. Therefore, simple drawing tools were added to the BeeSign 1.3 interface, supporting the option of temporarily hiding the students' predictions while the simulation was being run. It also appears from our observations that offering the students the option of choosing the color in which to make their prediction served a key motivational role as they often excitedly selected their colors.

In general, it appears that designing BeeSign to run on an interactive whiteboard to support group activity in this manner afforded the teachers the opportunity to lead productive group conversations in a very familiar manner. From the perspective of activity theory, this was a unique opportunity for the teachers to continue to engage with the students in the same kinds of rich discussions that they facilitate without the computer software, maintaining the division of labor and role of the community rather than leaving students to work individually and quietly at desktop computers as is required by more traditional applications.

Scaffolding Inquiry

There were two key features that appeared to support teachers and students in using BeeSign to engage in rich cycles of inquiry. First, BeeSign was designed to support incremental addition of complexity. A series of saved simulations were prepared that incrementally introduced new complexity into the interface, and new complexity into the inquiry discussions. For example, the first conversation about the bee behaviors only presented bees that do the bee dance, with only one active simulation window, so that students could focus their attention on the general patterns in how bees collect nectar. A later simulation introduced the idea of specifying that one hive dances and the other does not so that students could then compare and contrast the two. Just as important, the interface supported easy real-time experimentation, allowing response to student questions. For example, students would often ask how the bee behavior might change if one of the flowers was moved closer or further from the hive. The teacher could easily make this change by dragging the flower to a new location and then repeat the simulation to answer the question.

The second design feature to support robust inquiry was the inclusion of two simulation windows and a "match" feature. The two side-by-side simulation windows were intended to help students quickly see patterns in bee behaviors as they differed based on one or more variables. To support the teacher and students in organizing this productively, the interface includes a match button. The idea behind the match button is that the users can start by organizing one simulation window to reflect the features most important to the current experiment: the bee behavior, flower placement, flower variables such as nectar amount and quality, etc. Pressing the match button mirrors these important variables in the other simulation window. The user can then adjust one or more key variables to conduct an experiment. A frequent use of this included arranging the simulations, pressing match, and then adjusting the bee behavior on one window so that the observable differences between the two windows could be attributed to the bee behaviors and not other confounding features.

Prior research demonstrates that, left to their own devices, students typically do not design experiments that would adequately support control or comparison of a variable [8]. Not only was the teacher able to facilitate such controlled comparisons using BeeSign and the match button, but the students quickly came to request use of the match button on their own, evidencing successful scaffolding of their improved understanding of controlled comparisons.

Focusing on Patterns

A key decision in designing BeeSign was to ensure that the key qualitative patterns in the process through which honeybees collect nectar were easily visible to the students. In other words, the focus was not on quantitatively accurate modeling, as is the case with many other tools, but rather in supporting flexible but rich comparisons. As noted above, the dual simulation windows allowed students to quickly observe differences in bee behavior between them.

One key feature was that emerging patterns could typically be seen in several ways. For example, one can quickly note the difference in bee flight depicted in figure 1: the bees on the right fly directly to and from the flowers while the bees on the left continue to scatter in search of flowers (because they are not communicating via the dance). In fact, the students even named the pattern on the right a "chain" or "train" to help identify it. To complement this, students could also choose to represent the amount of nectar that was collected using the bar chart mentioned above. They then noticed the pattern in how bees that danced collected nectar more quickly. These two different views into the impact of the bee dance are not coincidental. They reflect two valuable ways of thinking about the results of the bee dance: impact on bee flight patterns as they search for nectar, and a resulting increase in nectar collection.

To further support students' observations of these aggregate-level patterns, the default view of the simulation is relatively close up (the bees and flowers are quite large). In implementing BeeSign 1.3, however, some students felt that the differences they observed were not dramatic enough to warrant a conclusion about the benefits of the bee dance. Therefore, BeeSign 1.3 included the option of zooming out. When zooming out, the simulation doesn't simply scale—it scales both the view and the range of the bees. Thus, flowers can be placed much further from the hive and yet the bees can still reach them. The zoom feature magnified the impact of the bee dance, helping many of the most skeptical students to see the pattern.

Introducing a Game

In the second implementation of BeeSign (1.3), we noted that there were several students who persisted in disputing the patterns in bee flight despite the fact that their peers had begun noting them quite regularly. An additional scaffold was introduced in the form of a highly motivational guessing game. With the guessing game feature, a simple click of the button randomly assigns one of the beehives to the dancing behavior, and one to the less efficient behavior of remembering a nectar source but not communicating it. Furthermore, the behavior titles are temporarily hidden so that it is not obvious before running the simulation which hive is engaging in which behavior. We then asked students to guess by raising their hand as soon as they felt they knew which hive was dancing, with an additional student serving as the "judge" of who raised their hand first. The students appeared to enjoy this activity, often finding it difficult to resist the urge to call out their prediction. More importantly, it supported rich discussion about the function of the bee dance as students paid attention to both the rate of nectar collection and the aggregate-level flight patterns.

CONCLUSIONS

Typically, curriculum and software designers assume that young children are not capable of rich, conceptually challenging, scientific activity. BeeSign, in contrast, was premised on the assumption that a thoughtfully designed software tool and supporting activity system can help engage these same young students in productive activities. The features and data presented here support this assumption. Specifically, BeeSign was designed and implemented with an intention of supporting small-group, teacher-led activity using an interactive whiteboard. This was complemented by features that assisted the students and teachers in engaging in cycles of inquiry. Finally, BeeSign was designed to make key aggregate-level patterns visible to the students who engaged with it in these cycles of inquiry. The results from the first implementation of BeeSign document that this was successful [1, 2]. Ongoing research with BeeSign 1.3 further demonstrates that students continue to be able to see and discuss rich patterns in honeybee behavior as a result of these interface choices.

Future research will further refine our understanding of the various design choices—those within the software directly and those that specify the nature of the activities to take place around the software—and how they support student learning of content in this context and others.

REFERENCES

1. Danish, J. A. BeeSign: A Computationally-Mediated Intervention to Examine K-1 Students' Representational Activities in the Context of Teaching Complex Systems Concepts. Dissertation, University of California at Los Angeles, Los Angeles, 2009.
2. Danish, J. A. BeeSign: a Design Experiment to Teach Kindergarten and First Grade Students About Honeybees From a Complex Systems Perspective. In Proceedings of the annual meeting of the American Educational Research Association (2009).
3. Goldstone, R. L. and Wilensky, U. Promoting Transfer by Grounding Complex Systems Principles. Journal of the Learning Sciences, 17, 4 (2008), 465-465.
4. Hmelo-Silver, C. E., Marathe, S. and Liu, L. Fish Swim, Rocks Sit, and Lungs Breathe: Expert-Novice Understanding of Complex Systems. Journal of the Learning Sciences, 16, 3 (2007), 307-331.
5. Jacobson, M. J. and Wilensky, U. Complex Systems in Education: Scientific and Educational Importance and Implications for the Learning Sciences. Journal of the Learning Sciences, 15, 1 (2006), 11-34.
6. Kaptelinin, V. and Nardi, B. A. Acting with Technology: Activity Theory and Interaction Design. MIT Press, 2006.
7. NRC Taking Science to School: Learning and Teaching Science in Grades K-8. National Academies Press, (Washington, DC, 2007).
8. Schauble, L. The development of scientific reasoning in knowledge-rich contexts. Developmental Psychology, 32, 1 (1996), 102-119.
9. Wilensky, U. and Resnick, M. Thinking in Levels: A Dynamic Systems Perspective to Making Sense of the World. Journal of Science Education and Technology, 8, 1 (1999), 3-19.