*Information Technology in Childhood Education Annual* (2002), 103-121

# Online Assistants in Children's Hypermedia Software

PENNY ANN GARCIA
*University of Wisconsin Oshkosh*
*Oshkosh, WI 54901, USA*
garcia@uwosh.edu

The classroom teacher's comfort and familiarity with computers and software influences student-computer use in the classroom. Teachers remain mired in repetitive introduction of basic software mechanics and rarely progress with students to advanced concepts or complex applications.

An Online Assistant (OLA) was developed to accompany the commercial hypermedia software, *HyperStudio*. This OLA introduced the software and provided a framework for students to storyboard stacks, sequentially introduced software mechanics, and suggested basic principles of good design. The impact of this OLA on the nature of classroom interactions and on the quality of projects produced was investigated. Control group students used the hypermedia software alone and the experimental group used the software with the OLA.

Projects employing the OLA were consistently rated higher than those using software alone. Student usage of the OLA significantly reduced teacher interactions dealing with basic software mechanics while significantly increasing interactions dealing with design, stack issues, and content. Student interactions significantly increased for the experimental group in the areas of advanced software issues, design, stack issues, and content. The OLA did not diminish the teacher's role but perceptibly shifted the nature of interactions to those areas requiring higher order cognition, and resulted in better hypermedia projects.

The use of computer technology in today's schools is often limited by educators who feel overwhelmed by the demands placed upon them to become proficient at the hardware, the software, and curriculum development simultaneously (Lockard, Abrams, & Many, 1997). The majority of educators are familiar with computer usage at a second state level of technological innovation (Naisbitt, 1982). At this level, the computer is used in conjunction with existing technologies to do what can already be done albeit more efficiently. In such a classroom, one finds teachers using computers as typewriters and using databases to keep student records and spreadsheets to keep grades. Students may be using integrated learning systems that dictate a learning path, employ skill and drill, and graph student successes and failures.

The majority of software developed over the past thirty years has reflected this second level of innovation. Computer assisted instruction has long focused on replicating the traditional classroom model of top-down teaching, drill and practice, and teacher directed learning (Uhr, 1969; O'Shea, 1979; Howe, O'Shea, & Plane, 1980). Later research began to acknowledge learning as a constructive process (Papert, 1980; Ennals, 1983; Glaser, Raghavan, & Shauble, 1988). Recent innovations in intelligent agents have pointed towards a collaborative partnership between computer and student (Maes, 1994).

New epistemologies of learning are emerging that indicate a dichotomy between theorists' accounts of cognition and the actual practices of people in learning activities (Resnick, 1987). It is increasingly evident that cognition does not reside entirely in the individual mind but may have portions of it distributed among other people and simple or complex tools (Pea, 1993). This research in distributed cognition, occurring over fifty years later, supports Dewey's (1938) statement that "Experience does not go on simply inside a person..." (p.39).

A subset of the current work on distributed cognition is off-loading, which occurs when a person places some of the cognitive burden for performing a task onto another person or tool (Salomon, 1993). This can be a positive condition such as when one creates a list of items rather than trying to carry those items in memory. However, it can be a negative experience in the classroom. Educators have long assumed portions of the student's cognitive burden when they make all the choices about what to study, how to study it, and what resources to use. In computer usage, this off-loading is part of what makes meaningful integrated computer usage so overwhelming to teachers. It is simply too much to be the expert on the hardware, the software, and the curriculum while simultaneously planning the implementation of the project with integrated computer support.

Given the rapid changes in computer software and hardware, teachers become increasingly frustrated as they attempt meaningful integration of computers into their curriculum. Additionally, the skill and drill software of yesterday does little to support today's constructivist teaching strategies. Instead, software applications that can serve as broad tools afford students and teachers wide opportunities for curriculum support by computers. The use of an online assistant, or OLT, can facilitate off-loading some of the teacher's responsibilities to the OLT smoothing the way for better computer integration into the learning experiences of students. With this support, students are able to assume responsibility for learning the mechanics of the software and for planning projects. This study focused on the addition of an online assistant to commercial hypermedia software, *HyperStudio*, and the subsequent impact on the quality of the completed project, the nature of teacher/student interactions, and the nature of student/student interactions.

## STUDY DESIGN

A quasi-experimental design was used in this study. Two existing, intact classrooms were used as the experimental and control groups. Random assignment of subjects to groups was precluded. Specifically a nonequivalent posttest-only control group design was employed. Both classrooms and their teachers were pretested to determine familiarity with the basics of the commercial software program to be used in the study. In addition, NCE scores on the Iowa Test of Basic Skills were compared between the two groups. This comparison as well as the software pretest determined initial equivalence of the experimental and control groups. The experimental group then received treatment (*HyperStudio* with OLT) while the control group used the software in its usual form unsupplemented by an OLT. Quality of the finished *HyperStudio* projects was assessed for both groups to determine effectiveness of the treatment.

Teacher/student and student/student interactions were evaluated for a sampling of five student pairs in both groups. Interactions in the following seven areas were tallied daily: Software issues - basic (defined as new cards, buttons, text boxes); software issues- advanced (defined as graphics, transitions, new button actions); design issues (defined as color, font, layout); stack issues (defined as linking and layout); spelling and grammar; content material (defined as what content was appropriate); and, content composition (defined as how to state content appropriately for hypermedia).

Subjects participating in the study were 45 fourth-grade students from an urban area elementary school. There were 23 students in the group using

*HyperStudio* with OLT and 22 students in the group using *HyperStudio* alone. An examination of the Iowa Test of Basic Skills National Percentile Rankings, administered in third grade, determined that they were of similar achievement abilities. Additionally, in a presurvey to determine prior knowledge of the software, each group indicated that they had only had limited exposure in third grade to the software. Neither group had attempted a project using the software during fourth-grade prior to this study.

Hypermedia projects completed during the three week data collection period were collected from both groups, copies made, and distributed to raters who used an evaluation tool, the MEDIA HCI Design Evaluation (Druin, 1998), that assessed four areas. The first area, *Reaction to Visual Design*, assessed the stack according to card layout and design, the organization of information, and the sequence of cards from the user's standpoint. The second, *Reaction to Stack Construction*, examined the user interface with the stack as to whether the user found the stack interesting, easy to navigate, and believed that user needs had been served. *Reaction to Content*, the third evaluation area, evaluated the actual content material covered in the stack for readability, accuracy, and appropriateness of discourse form to hypermedia format. Finally, the user's *Overall Reaction to Stack* was assessed.

The teacher of the classroom using the software with OLT was interviewed to discuss her reactions to the presence of the online assistant, how effective she felt it was with her students, and her general reactions to helping students develop a project using *HyperStudio*. Additionally, an exit interview was held with the teacher of the classroom that used the software alone to discuss her reactions to helping students develop a project using *HyperStudio*. Students were surveyed to assess their acceptance of the OLA.

## TREATMENT

Both the experimental group and the control group used *HyperStudio* to develop a project that was a stack consisting of a minimum of five cards. Both groups had access to the commercial software *HyperStudio* and, for assistance in their project, their classroom teacher. However, the experimental group also had access to an OLT developed by the researcher to assist students in planning their project, in using the *HyperStudio* software, and in developing awareness of rudimentary design elements. The content material for each group was determined by the teacher of that group based on curricular needs at the time of the study. Thus, the group using the OLT developed stacks on states in the southeast region of the United States reflecting their current social studies curriculum while the group using the software alone developed stacks on weather for their current science curriculum.

The researcher envisioned and developed an OLT that assisted the students in designing a *HyperStudio* stack by introducing a storyboard approach to design. This component of the OLT then sent the students offline to develop a storyboard using index cards and yarn for links. Students then consulted with their teacher for a discussion on both links and the card content. After consensus was reached between the teacher and student on the storyboard, cards and links were finalized. This storyboard served as the working plan to be followed when students returned to the software to begin stack construction.

Upon returning to the software to begin stack development, the OLT sequentially guided the student through a series of steps. First, students created each card with only text boxes and content inserted in those boxes. Next, the portion of the OLT was brought online that assisted students through the steps of adding buttons. After students had demonstrated their stack to their teacher, as directed by the software, the next portion of the OLT was added. This portion assisted students in adding graphics to their existing cards. Finally, the OLT offered assistance on modifying existing text and button objects. Concurrent with the online components that addressed adding buttons, adding graphics, and modifying objects, the availability of design hints were added. These design hints covered such items as choice and variety of fonts, color selection, and button transitions.

Addy, a dog that already periodically appears in existing dialogue boxes in *HyperStudio*, was selected as the persona for this OLT. The researcher felt that the appearance of Addy both in dialogue boxes and as the OLT would maintain consistency within the software. The name of the OLT service was *Ask Addy*. As the OLT, Addy offered assistance in three areas: design and content, software application instructions, and design considerations.

The researcher developed the OLT using *HyperStudio* software. The OLT was implemented as a series of independent stacks, each linked to a menu card that was in turn linked to the cards in the students' stack. This enabled the researcher to bring various components of the OLT online sequentially throughout the experimental period. To manage memory size, the OLT contained only text and graphics. No voice or animation features were included. This sequential introduction of the OLT enabled a certain amount of control over the students' process.

## RESEARCH RESULTS

### Can an OLT, used in conjunction with application software, impact the quality of students' completed hypermedia projects?

There was a significant difference between the holistic quality scores of the completed projects of the group using software with the OLT and the quality scores of the group using software alone. The results of the MEDIA HCI Design Evaluation forms, completed by raters on completed *HyperStudio* stacks at the end of the project period, showed that the mean holistic score of the stacks completed by students in the group using the OLT was 38.9 points higher than the mean of those students using the software alone. Therefore, results of the study suggest that the presence of an OLT significantly impacts the quality of students' completed hypermedia projects.

**Table 1**

*T* Test Analysis of Students' Completed Hypermedia Projects

| | Means | *SD* | *t* value | Significance (2 tailed) |
|---|---|---|---|---|
| With Online assistant | 100.60 | 16.24 | 5.143 | 000* |
| Software alone | 61.10 | 18.06 | | |

*$p<.05$

### Does the use of an OLT during the project alter the nature of teacher/student interactions?

To measure the *nature* of teacher/student interactions, seven areas were chosen for consideration. These seven areas formed a hierarchy of possible interactions. At a rudimentary level, the frequency of interactions concerning basic software issues about mechanics of the *HyperStudio* software and spelling and grammar was measured. Then interactions about advanced software skills such as importing graphics and specialized button actions were measured. Next, interactions were measured that required the application of more complex thinking skills such as design issues, stack issues, content material, and content composition. Finally, the sum of the total number of interactions in all seven areas was computed and analyzed.

There was no significant difference in the total number of interactions between teacher/students in the group using software with the OLT and the group using the software alone. However, a shift in the *nature* of teacher/student interactions could be seen as indicated by the frequencies of interactions occurring at the differing levels of complexity. The observed pairs in the group using software with the OLT had significantly fewer interactions with their teacher about basic software issues. There were significantly more interactions between teacher and students in the group using software

with the OLT in the complex areas of design issues, stack issues, content material, and content composition. Thus, results of the study suggest that the presence of an OLT positively impacts teacher/student interactions by increasing the number of interactions concerning areas that require higher-order thinking skills and reducing the number of interactions concerning basic software issues.

**Table 2**

*T* Test Analysis of Teacher/Student Interactions

| | Means | *SD* | *t* value | Significance (2-tailed) |
|---|---|---|---|---|
| Basic Software Issues | | | | |
| With Online Assistant | 10.00 | 3.16 | 3.82 | .005* |
| Software Only | 20.00 | 10.65 | | |
| Advanced Software Issues | | | | |
| With Online Assistant | 11.80 | 3.70 | .925 | .381 |
| Software Only | 13.80 | 3.12 | | |
| Design Issues | | | | |
| With Online Assistant | 21.00 | 2.35 | 6.178 | .000* |
| Software Only | 8.40 | 3.91 | | |
| Stack Issues | | | | |
| With Online Assistant | 35.40 | 3.40 | 8.984 | .000* |
| Software Only | 5.20 | 1.50 | | |
| Spelling and Grammar | | | | |
| With Online Assistant | 4.20 | 1.49 | .577 | .580 |
| Software Only | 4.80 | 1.79 | | |
| Content Material | | | | |
| With Online Assistant | 37.60 | 7.09 | 9.774 | .000* |
| Software Only | 5.60 | 1.82 | | |
| Content Composition | | | | |
| With Online Assistant | 28.20 | 7.16 | 6.640 | .000* |
| Software Only | 5.80 | 2.39 | | |
| Total Interaction | | | | |
| With Online Assistant | 105.86 | 65.27 | 1.815 | .095 |
| Software Only | 51.86 | 43.98 | | |

*$p$<.05

**Does the use of an OLT during the project alter the nature of student/student interactions?**

To measure the *nature* of student/student interactions, seven areas were chosen for consideration. These seven areas formed a hierarchy of possible interactions. At a rudimentary level, the frequency of interactions concerning basic software issues about mechanics of the *HyperStudio* software and

spelling and grammar were measured. Then interactions about advanced software skills such as importing graphics and specialized button actions were measured. Next, interactions were measured that required the application of more complex thinking skills such as design issues, stack issues, content material, and content composition. Finally, the sum of the total number of interactions in all seven areas was computed and analyzed.

There was no significant difference in the total number of interactions between student/student in the group using software with the OLT and the group using the software alone. However, a shift in the *nature* of student/student interactions could be seen as indicated by the frequencies of interactions occurring at the differing levels of complexity. There were significantly more interactions between students in the observed pairs in the group using software with the OLT in the areas of advanced software issues, design issues, stack issues, and content issues, which require the application of complex thinking skills. Thus, results of the study suggest that the presence of an OLT positively impacts student/student interactions by increasing the number of interactions between students concerning areas that require higher-order thinking skills.

**Table 3**

*T* Test Analysis of Student/Student Interactions

| | **Means** | ***SD*** | ***t* value** | **Significance (2-tailed)** |
|---|---|---|---|---|
| Basic Software Issues | | | | |
| With Online Assistant | 48.00 | 9.83 | 1.778 | .126 |
| Software Only | 35.25 | 10.44 | | |
| Advanced Software Issues | | | | |
| With Online Assistant | 54.50 | 6.35 | 4.047 | .007* |
| Software Only | 38.75 | 4.50 | | |
| Design Issues | | | | |
| With Online Assistant | 47.25 | 7.80 | 3.097 | .021* |
| Software Only | 32.25 | 5.74 | | |
| Stack Issues | | | | |
| With Online Assistant | 33.75 | 7.27 | 4.773 | .003* |
| Software Only | 13.00 | 4.76 | | |
| Spelling and Grammar | | | | |
| With Online Assistant | 16.50 | 9.11 | .233 | .824 |
| Software Only | 14.75 | 11.95 | | |
| Content Material | | | | |
| With Online Assistant | 71.25 | 11.32 | 4.647 | .004* |
| Software Only | 41.50 | 5.97 | | |
| Content Composition | | | | |
| With Online Assistant | 29.25 | 8.96 | .210 | .841 |
| Software Only | 30.50 | 7.85 | | |
| Total Interaction | | | | |
| With Online Assistant | 171.71 | 72.12 | 1.747 | .104 |
| Software Only | 115.57 | 44.14 | | |

*$p<.05$

**How do students self assess the presence and usefulness of an OLT?**

All students in the group using software with the OLT completed a Posttreatment Survey. From the responses on this survey it may be concluded that a majority of students found *Ask Addy* to be helpful, easy to understand, and quick to get help. A majority of students would like *Ask Addy* to be able to offer more help than was available in the prototype. However, the majority of students reported that they still had to ask their teacher for help even after using *Ask Addy* and the majority of students felt that they turned to *Ask Addy* less often than they did their teacher. Over 50% of the students indicated that *Ask Addy* was most helpful in helping them learn how to design a *HyperStudio* stack. Secondarily, *Ask Addy* was most helpful in assisting students in learning how to make text boxes. Thus, results of the study suggest that the presence of an OLT is accepted by students as most useful in helping design a hypermedia project and as an acceptable source for information.

**Table 4**

Student Self-Assessment of Reaction to Presence and Usefulness of the OLA

| | Not at all 1 | 2 | 3 | 4 | Very 5 |
|---|---|---|---|---|---|
| I found *Ask Addy* helpful. | | 9% | 39% | 48% | 4% |
| I used *Ask Addy* more often than I asked my teacher for help. | 13% | 44% | 17% | 13% | 13% |
| I would like *Ask Addy* to be able to offer even more help. | 22% | 17% | 9% | 13% | 39% |
| *Ask Addy* was easy to understand. | | 9% | 13% | 26% | 52% |
| *Ask Addy* made it quick to get help. | | 26% | 31% | 39% | 4% |
| I often had to still ask my teacher for help after using *Ask Addy* | 4% | 17% | 22% | 48% | 9% |
| Percentage of Students Who Indicated That… | | | | | |
| "*Ask Addy* helped me most to…" | | | | | |
| …learn about how to design a *HyperStudio* stack | 52% | | | | |
| …learn how to make text boxes | 26% | | | | |
| …learn how to make buttons | 9% | | | | |
| …learn how to make my cards look good | 13% | | | | |

**What is the relationship between student acceptance of the OLT and the quality of their completed hypermedia stack?**

Data analysis confirmed that there is a moderate correlation between student acceptance of the OLT and the quality of their completed hypermedia stack. This acceptance of the OLT was indicated by 90% of the students indicating that they found *Ask Addy* helpful. Over 50% of the students also indicated that they would like to be able to use *Ask Addy* for getting more help. Students indicated that they found *Ask Addy* to be most helpful in the design process and in adding text boxes. This overwhelming student acceptance of Addy's assistance positively correlated with raters' assessment of the quality of these same students' stacks. Thus, results of the study suggest that there is a positive relationship between student acceptance of an OLT and the quality of their completed hypermedia project.

**Table 5**

Pearson's Stack Quality and Student Acceptance of the OLA Correlation

| | | Student Reaction | Stack Quality |
|---|---|---|---|
| Pearson Correlation | Student Reaction | 1.000 | .573 |
| | Stack Quality | .573 | 1.000 |
| Significance (2-tailed) | Student Reaction | | .004* |
| | Stack Quality | .004* | |

*$p$<.01

## DISCUSSION OF RESULTS

The completed projects of the students in the group using software with the OLT were consistently scored higher by raters than those completed by students in the group using software alone. The highest scoring for the group using software with the OLT was in the areas of stack construction, visual design, and overall reaction to the stack.

Although both groups of students had been instructed by their teacher to create a minimum five-card stack, those students in the group using software with the OLT went well beyond that five-card minimum. Rater 1 noted, "...generally, the stacks on the states (by students using OLT) had many more cards than the stacks on weather...and, usually, these stacks were logically connected showing an inter-connected web-like pattern." It was noticed by Rater 3 that "...weather stacks (done by group with software alone) were more linear in nature."

The greater number of cards in these stacks and the more web-like construction may be attributed to the storyboarding process. The process of storyboarding supports constructivist methodology—emphasizing problem-solving—while simultaneously placing back on the student responsibilities formerly off-loaded to the teacher. The researcher observed that the idea of storyboarding did not occur naturally to most students. Indeed, only one pair of students in the group using software alone blocked out their stack design before beginning the production period. As the production period progressed, this group lost sight of their original plan and, after the first week, did not refer back to it. In contrast, the group that was referred offline by the OLT to develop storyboards spent three to four days on the task and most students referred to them on a daily basis. Observed pairs of students in the group using the OLT often interacted about button placement and content material through the use of their storyboards.

The idea of having the students storyboard the project did not occur naturally to the teacher of either group. The teacher of the group using the software alone concluded,

> I'm really glad my students had the opportunity to complete this project but (the teacher of the group using OLT) showed me her stacks and my kids just did the minimum. I should have had them do more planning of the stack before they started their projects. They had all their research done but they needed to give more thought to how their stack would be made.

The teacher of the group using the OLT noted that although she used pre-writing strategies in Writers' Workshop she "…never really saw the need for strategies before students sit at a computer."

The teacher of the group using the OLT also said,

> I was pleasantly surprised by the fact that the five-card minimum was never a discussion point with these kids. They got all carried away with their storyboards and connections and put down so many more cards. I thought they'd never complete it but since they had this plan (storyboard) that *they had committed to*, they did it!

It is important to note the italics in the prior quote. The students had committed to the plan because *they* created it. The teacher did not impose the plan; the product was not the emphasis. The storyboard places the emphasis on the process instead. This is consistent with constructivist methodology (Roblyer, Edwards, & Havriluk, 1996). Additionally, the creation of a storyboard allowed the students to assume responsibilities or *executive*

*functions* (Perkins, 1993)—deciding on content, locating resources, arrangement of those resources, appropriate number of cards for presentation of content—that would have traditionally been offloaded to the teacher.

Students in the group using the OLT indicated that they were able to assume these responsibilities if given the opportunity. Fifty-two percent of the students in this group believed that *Ask Addy* helped them most when learning about designing a *HyperStudio* stack. Before showing students a sample of *HyperStudio* storyboarding, *Ask Addy* pointed out that most thought patterns form a web rather than a linear design. The students were then directed offline to create these storyboards. This offline period lasted from three to five days, depending on the student group.

This period of time provided a rich environment for teacher/student and student/student interactions on the content of the material to be presented in the stack. The group using the OLT showed a significantly higher number of teacher/student and student/student interactions about content material. The bulk of these interactions occurred during the storyboarding period. The researcher's journal recorded one of these observed student/student interactions:

> Student 1B: Pictures (of the state symbols) aren't enough. We have to say something about them, too.
>
> Student 1A: (Teacher) didn't say we couldn't just use pictures. I can draw the bird good…
>
> Student 1B: You draw the bird but I want to write about them too. Find me that book we used in class…I think it will tell us more…
>
> Student 1A: OK…Ok…but I don't think that book is good…I'll look in the library for another (book).

Here two students push themselves to go further than they originally perceived as necessary. The content is rudimentary but evidently it is important to at least one student to have more than just pictures and both are willing to go locate other resources for the information. *Ask Addy* did not generate this interaction but it did contribute to an environment in which it occurred. Students from both groups came to the production period with their research well under way. However, students from the group using the OLT were most willing to modify that content, either expanding it or cut-

ting it, during the storyboard process. Another interaction recorded by the researcher shows the winnowing process that occurred as well:

> Student 3B: We have way too much writing about places to visit here. I don't want to have to type that much on the computer.
>
> Student 3A: Addy said that this thing (stack) should be like a museum…
>
> Student 3B: I don't think anyone would even read all of this stuff.
>
> Student 3A: Are there pictures of some of these places? In the books we used?

At this point, the conversation was terminated due to the end of the class period. However, it was not forgotten because on the next day, the issue of "places to visit" came up again in a different context.

> Teacher 1: Did you get a map completed?
>
> Student 3A: (Student B) did it in class yesterday. Do we have to draw it again on the computer?
>
> Student 3B: No…I think we can use the scanner instead.
>
> Teacher 1: Did you have all the information (on the map) completed?
>
> Student 3B: Yeah, the capital, the rivers, some of the places to visit…
>
> Teacher 1: Places to visit…are you going to link these to this section over here? (pointing to storyboard plan)
>
> Student 3A: Hey, we could…with just pictures…then we could cut out some of that writing…
>
> Student 3B: So we just make buttons on the map that they click on and see pictures of places to go visit?
>
> Student 3A: Right…but we'd still have to probably write something about them…

In these conversations, it is apparent that *Ask Addy* did make the students think about the discourse form appropriate in hypermedia environments. Although the student's first concern about too much writing was based on the thought of having to "type" it all in, the issue shifted to a design one and how to appropriately present the material in a hypermedia context. The teacher/student interaction was a gentle one with the teacher using the storyboard to suggest a linkage between two areas. This also shows that the storyboarding process provided ample opportunities for the teacher and students to talk about how their stack was constructed and how things were linked. The *t* test analysis of stack issue interactions between teacher and student showed a significance difference. The teacher of the group using the OLT averaged 37 interactions with students on the stack construction and layout while the teacher of the group using the software alone averaged only five interactions with students. This lack of interaction was reflected in the smaller stack size and linear design of the students using the software alone.

*Ask Addy* was able to significantly reduce the number of teacher/student interactions that dealt with basic software issues. Basic software issues included the mundane mechanical processes of the *HyperStudio* software such as creating new cards, adding text boxes, and creating buttons to link cards. These mechanics form the foundation for all work in *HyperStudio* and are relatively easy to learn and yet require much repetition from the classroom teacher if a one-to-one teaching methodology is employed. The teacher of the group using software alone was heard to mutter, "How many times do I have to show them (the students) the same thing (how to create a textbox)?" The teacher of the group using software with the OLT early in the project period mirrored this exasperation. However, she stated in the exit interview, "I had to get used to sending them (the students) to *Ask Addy*. Their first inclination is always to ask me.... and I just naturally move in to help. I guess it took me a while to get used to the fact that I didn't have to have all the answers but I did have to help them remember to use *Ask Addy*."

Equally important is the sequential manner in which the OLT was brought online. Total facility with any software is a matter of practice and experience. Educators usually lack the time to develop that facility and are therefore hesitant to move students ahead in a project using unfamiliar software. Not knowing in what sequence to introduce software mechanics can frustrate teachers. The teacher of the group using software alone commented on the third day of production, "Text boxes, graphics, scanning, buttons, sound…too many questions from too many directions."

*Ask Addy* was brought on line sequentially in a manner that facilitated students' completion of their storyboard plan. Initially, *Ask Addy* only provided help in creating new cards and adding text boxes in which to enter

text. *Ask Addy* asked that students make all the cards in the stack first before adding buttons and graphics. Because this involved the bulk of the online work (word processing) the teacher of this group became concerned. "Will they actually finish this project?" she asked, eight days into the data collection period. Having spent three to four days storyboarding and then another four to five days creating the rudiments of the cards for the stack, it did indeed seem that they were lagging behind. However, the finished products demonstrated that the sequential introduction of skills facilitated that completion of the stacks as originally envisioned by the student.

It is important to note from the data analysis of teacher/student interactions that the addition of an OLT did not significantly affect the total number of teacher/student interactions. However, it did impact the nature of the teacher/student interactions. As previously discussed, the storyboarding period required by *Ask Addy* did significantly increase the number of teacher/student interactions related to stack issues, and content material. In addition, there were significantly more teacher/student interactions concerning content composition and design issues during the time spent online with *HyperStudio* stack development.

Students' self-assessments of the presence of the OLT also supported the conclusion that the need for a teacher is not diminished by the online assistance. Fifty-seven percent of the students indicated that they did not *Ask Addy* for help more than they asked it of their teacher. The same percent of students also reported having to ask their teacher for help even after asking Addy for assistance.

The total number of student/student interactions was also not significantly different in observed pairs in both groups. However, the nature of that interaction was different as observed in the frequency of interactions in the areas of advanced software issues, design issues, stack issues, and content material. As previously discussed, the story boarding period provided an environment in which students could have intense discussions on stack design and content material. During the actual stack production period on the computer, student/student interactions gradually shifted to design issues and advanced software issues.

*Ask Addy* moved students quickly along in basic software mechanics. Students in the group using software with an OLT expressed more curiosity and interest in importing graphics and clip art to cards, working with sound and *new button actions*. For example, student pair four was interested in making their timeline more interesting. They asked their teacher for assistance who in turn correctly referred them to the *Ask Addy* section on *new button actions*. Looking down the list of possibilities, they became intrigued

with "hide-show." Intrigued with the example Addy provided them, they spent two class periods incorporating that button action into their timeline so that as dates were clicked on, text boxes popped up telling what happened on that date.

Students in the group using *HyperStudio* software only tended to draw directly on the screen using the *HyperStudio* draw tools. Few pairs attempted to scan pictures to use in their stacks. Rater 2 noted "…(this stack, weather) used exclusively student drawn graphics," "…(this stack, weather) was one of the few on this subject that scanned in a commercial picture." By contrast, the students in the group using software with the OLT used a wider variety of methods for importing pictures to the screen. Rater 2 made the following comments: "this stack (states) has students who were reluctant to draw directly onto the computer screen," "…(this stack, states) drew picture by hand and then scanned it into program," and "…and, stacks on states, used more commercial pictures scanned in." Two of the three raters remarked that the stacks on weather were "charming," "childlike," and "more original."

This debate on whether to draw original pictures or bring in commercial pictures, and whether to use a patterned or plain background provided much of the content for student/student interaction on design issues. *Ask Addy* gave the students a few general design pointers: cards in stacks should look like they belong in the same family, font colors and background colors should be reviewed for readability, too many font styles may distract the reader, and so forth. Students read these design tips and then debated over whether to use them or have fun with the possibilities of the program. One student said to his partner, "I don't care what Addy says, I like lots of colors." Another comment was made, "I think those colors look cool…I don't care if they are hard to read." One student, upon viewing the sample card showing over-use of fonts compared with conservative use of fonts remarked, "But that one (many font styles) looks better than that one (sample with only two font style)!" It seems that while *Ask Addy* stimulated discussion of good design principles and raised students' awareness of the need to consider them, it certainly did not limit students' natural curiosity and experimentation.

Some students emerged as resident experts on certain aspects of software mechanics. For example, as the researcher observed some groups reaching the point where they would be ready to begin linking the cards, the third part of *Ask Addy* that demonstrated how students could add buttons was brought online. At this juncture, an interesting shift occurred. One group of students (on a computer named "Wilbur") finished all their cards

particularly early and was eager to begin adding buttons. They were pleased to find the third part of *Ask Addy* available for their use that morning. This group noisily made use of the OLT with much discussion between them. Getting the stack connected took them only one class period and they rapidly became experts on the process. The researcher observed that of the nine remaining groups of students who needed to learn how to add buttons, only four groups made use of the OLT exclusively. The remaining five groups divided up with one student going through the *Ask Addy* screen while the other student went to the "Wilbur" group (the first group to use *Ask Addy* to add buttons) to observe them and ask questions about buttons. *Ask Addy* provided immediate assistance to the group that finished first so that they could proceed with their project without having to wait for whole-group instruction or relying on the teacher to provide one-on-one instruction. This group then became recognized by their peers as the resident-experts on adding buttons, which resulted in an impromptu *Jigsaw II* (Slavin, 1990) method emerging. Again, this emerged from the student process. It did not require that the teacher assume the responsibility for building in the peer tutoring as part of the process.

Student responses on the Posttreatment Survey when correlated with the MEDIA HCI Design Evaluation Form holistic scores indicated a moderate correlation between student acceptance of the OLT and the quality of the finished project. Students demonstrated a purposeful acceptance of *Ask Addy*. The majority of students indicated that they did find *Ask Addy* helpful, particularly in the areas of design and adding textboxes. These two areas are indicative of student concern with the complex issues of content and process. The *Ask Addy* prototype was devoid of anthropomorphic qualities; Addy's assistance was straightforward and presented in a formal manner. Therefore, students' acceptance of *Ask Addy* was based on the quality of help offered rather than an attraction to the character, Addy.

Students did indicate a fascination with the possibilities of an OLT. In discussions throughout the production period, they discussed their visions of what *Ask Addy* could be. Students wanted much more than what the prototype could offer. They wanted *Ask Addy*'s bank of assistance to be much broader. A more computer literate student suggested, "Why doesn't *Ask Addy* just let us type in a word to get help like on my computer at home?" Querying this student, it was determined that he was using *Microsoft's Office 97* software at home where he could "ask the paper clip."

One student expressed frustration that she couldn't practice the skill being explained while in the *Ask Addy* screen. Another suggested, "I wish *Ask Addy* would stay around when I got back to my stack…I forget what she says by the time I get here." This leads to consideration of adding an

electronic white board space to the *Ask Addy* software where a student could practice while online with the tutor.

As a prototype, *Ask Addy* was purposefully designed to be devoid of any character skills. However, another student compared Addy to the assistance she can receive in a video game on her home system. She indicated that she would like to have the option to "choose any character I want to be my helper instead of just having a dog." This would indicate an appropriate juncture for prototyping an OLT as an intelligent agent who is anthropomorphic with certain personality traits.

## CONCLUSION

An OLT can help students and teachers facilitate a process and produce better quality work. It can also suggest a learning environment that is conducive to a different, more complex nature of teacher/student interactions. While the act of directing students to make a storyboard prior to sitting at the computer to begin the production process did not require an OLT built into software, it demonstrates that an OLT can provide valuable guidance in a suggested process for using that software. This would help a teacher who is unfamiliar with the basic hypermedia software in the implementation of a project.

The need for a teacher is not diminished by the presence of an OLT. Instead, the role of the teacher is perceptibly altered in the interactions between teacher and student. The online assistant is able to assist the students in learning the rudimentary mechanics of a hypermedia software program. It can make them more aware of design issues and discourse forms peculiar to hypermedia. It can stimulate children to take more responsibility for those higher level activities such as planning strategies, locating resources, and organizing information that have typically been offloaded onto the teacher. In this way, teachers are able to interact with students concerning more complex content and process issues while guiding their discovery and development by directing them to an online assistant.

## References

Dewey, J. (1963) (Originally published 1938). *Experience and education.* New York: Macmillan.

Druin, A. (1998). *The MEDIA program.* Albuquerque, NM: University of New Mexico.

Ennals, R. (1983). *Beginning micro-prolog*. New York: Harper & Row.

Glaser, R., Raghavan, K., & Schauble, L. (1988). VOLTAVILLE, a discovery environment to explore the laws of DC circuits. In G. Gauthier & C. Frasson (Eds.), *Proceedings of intelligent tutoring systems-88* (pp. 61-67), University of Montreal, Canada.

Howe, J.A.M., O'Shea, T., & Plane, F. (1980). Teaching mathematics through LOGO programming: An evaluation study. In R. Lewis & E.D. Tagg (Eds.), *Computer assisted learning: Scope, progress and limits* (pp. 85-102). Amsterdam, NY: North Holland.

Lockard, J., Abrams, P.D., & Many, W.A. (1997). *Microcomputers for twenty-first century educators* (4th ed.). New York: Longman.

Maes, P. (1994). Agents that reduce work and information overload. *Communications of the ACM, 35*(7), 31-41.

Naisbitt, J. (1982). *Megatrends: Ten new directions transforming our lives*. New York: Warner Books.

O'Shea, T. (1979). *Self-improving teaching systems: An application of artificial intelligence to computer assisted instruction*. Basel, Switzerland: Birkhauser Verlag.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas.* New York: Basic Books.

Pea, R.D. (1993). Distributed intelligence and designs for education. In G. Salomon (Ed.), *Distributed cognitions: Psychological and educational considerations* (pp. 47-87). Cambridge, UK: Cambridge University Press.

Perkins, D.N. (1993). Person-lous. In G. Salomon (Ed.), *Distributed cognitions: Psychological and educational considerations* (pp. 88-110). Cambridge, UK: Cambridge University Press.

Resnick, L. (1987). Learning in school and out. *Educational Researcher, 16*(9), 3-21.

Roblyer, M.D., Edwards, J., & Havriluk, A. (1996). *Integrating educational technology into teaching*. Englewood Cliffs, NJ: Prentice Hall.

Salomon, G. (Ed.). (1993). *Distributed cognitions: Psychological and educational considerations*. Cambridge, UK: Cambridge University Press.

Slavin, R. (1990). *Cooperative learning, theory, research, and practice*. Englewood Cliffs, NJ: Prentice Hall.

Uhr, L. (1969). Teaching machine programs that generate problems as a function of interaction with students. In *Proceedings of the 24th National ACM Conference* (pp. 125-134). New York: Association for Computing Machinery.