Liu, L., & Maddux, C. (2010). Using dynamic design in the integration of Type II applications: Effectiveness, strategies and methods. *International Journal of Technology in Teaching and Learning, 6*(1), *71-88.*

Using Dynamic Design in the Integration of Type II Applications: Effectiveness, Strategies and Methods

Leping Liu & Cleborne Maddux
University of Nevada, Reno

Dynamic design principles have been found to be useful in designing a variety of educational applications. Type II applications have been widely used to assist teaching and learning. In this article, the authors discuss a way of using dynamic design in the integration of Type II applications in education. The article first reviews the features of Type II applications, and introduces an approach that characterizes design as s*tatic design* or *dynamic design*. An exploratory study is reported that examines the effectiveness of using a Type II, dynamic design to improve the design of online computer based instructions. Examples using this design approach to improve the quality of learning in different areas will be presented.

Keywords: dynamic design, static design, Type II application, storyboarding, Logo, geometric learning, online communications, online counseling, course management

INTRODUCTION

Educational computing applications have been classified as either *Type I* or *Type II*, and this classification has been used in the field of information technology in education for decades (Maddux & Johnson, 2006; Liu & Jones, 2008). A *Type I application* is defined as "the use of technologies that make it easier, quicker, more efficient, or more convenient to learn or teach in traditional ways;" while a *Type II application* is "the use of technologies that make available new and better ways of learning and teaching" (Maddux, Johnson, & Willis, 2001, p. 119). While there is nothing wrong with making use of Type I applications in education, Type II applications are more likely to lead to the accomplishment of more important educational goals such as the promotion of problem-solving skills and the promotion of higher order thinking skills in general. Although a balanced use of both Type I and Type II applications is necessary to the success of

Leping Liu, Professor, and Cleborne Maddux, Foundation Professor, are at the Department of Counseling and Educational Psychology, University of Nevada, Reno. Please contact Dr. Liu at liu@unr.edu.

technology integration (Maddux, 2003; Parr, 2002), the history of technology integration has been dominated by Type I applications. Type II applications have been far less common (Johnson, Maddux, & Liu, 1997, 2000; Liu, Maddux, & Johnson, 2008). This is unfortunate, since Type II applications involve more active user intellectual involvement, and promote more creative tasks than do Type I applications (Maddux, Johnson, & Willis, 2001, p. 119). The integration of Type II applications is generally more challenging than integration of Type I applications and requires a careful and systematic design. Unfortunately, the design of integration has been a weak area in both research and practice (Liu &Velasquez-Bryant, 2003; Liu, Jones, & Sadera, 2010).

The current authors have proposed a new approach to design that differentiates between *static design* and *dynamic design* (Liu, 2003; Liu & Johnson, 2004). Use of dynamic design principles has been found to have a positive impact on the design of technology-based learning (Liu & Johnson, 2004, Liu & Maddux, 2005). However, no studies have been conducted to explore how dynamic design principles could be applied to the integration of Type II applications in education. The purpose of this article is to demonstrate how the concept of Type II applications can be combined with the concept of dynamic design to facilitate the integration of technology in education.

In the following sections, the authors will first review the features of Type II applications, and discuss both static and dynamic design. Next, an exploratory study will be reported that examines the effectiveness of using a Type II storyboarding application tool, with dynamic design, to improve the design of online computer based instructions. Finally, a set of examples will be presented in which this new approach to design is used to improve the quality of different learning experiences.

FEATURES OF TYPE II APPLICATIONS

Type II applications of information technology in education are those that lead to new and better ways of teaching and learning. Maddux, Johnson, and Willis (2001) discussed five features shared by most Type II applications:

1. They require relatively active user intellectual involvement.
2. They place much of the control of what happens on the screen in the hands of the user.
3. They give the user control of the interaction between user and machine.
4. The goal of the application is the accomplishment of relatively creative tasks.
5. Many hours of use are required before the user experiences everything the software is capable of doing. (p. 119).

Information technology applications that have all or most of the five features can be considered Type II applications. Of course merely selecting a Type II application does not guarantee a positive educational outcome. Once a Type II application is identified and selected, the next issue is whether it can be used appropriately to achieve specific learning goals. This will depend upon the design of integration.

STATIC DESIGN AND DYNAMIC DESIGN

Over the years, the ADDIE (Analysis, Design, Development, Implementation, and Evaluation) instructional design model has served as a useful framework for integration design. ADDIE involves specific procedures and tasks of design at each of several stages (Reiser & Dempsey, 2007; Smith & Ragan, 2005). Successful strategies and methods for use of the model differ depending on the context of each case and need to be determined in advance. A new approach that categorizes design as either static or dynamic has been useful in thinking about such strategies and methods (Liu, 2003; Liu & Johnson, 2004, Liu & Maddux, 2005). The type of design determines the rules, principles, structures, or macro processes of educational designs such as program design, course design, instructional design, technology integration design, or design of any educational applications.

Static design has four characteristics:

1. It is a *linear* design. All units proceed one after another through one single "path."
2. It is a *single dimensional* design, emphasizing one major aspect of the application.
3. It focuses on the *state* of each individual unit. For example, it may concentrate on the weight or feature of each unit.
4. It is a *close-ended* design without room for change.

Dynamic design has four different characteristics:

1. It is *nonlinear*. All units can be in a "tree" or "net" structure and do not have to be performed one after another.
2. It is a *multiple dimensional* design, emphasizing interactions among all or most aspects of the application.
3. It focuses on continuous *process* of all units. It emphasizes the connections among them.
4. It is an *open-ended* design with developmental potential.

Table 1 summarizes the features of *Static Design* and *Dynamic Design*. A design can be categorized as a static design or dynamic design if it meets any one feature or a combination of certain features under that type.

Table 1. Two Types of Design

Features	**Static Design**	**Dynamic Design**
1	Linear	Non-linear
2	Single-dimensional	Multiple-dimensional
3	State-focused	Process-focused
4	Close-ended	Open-ended

All Type II applications are not necessarily based upon dynamic designs. However, research suggests that using dynamic design often results in higher achievement and more positive learning outcomes (Liu, 2003; Liu & Johnson, 2004; Liu & Maddux, 2005). The following study is one example of the application of dynamic design principles to the integration of a Type II application. Specifically, this example deals with the design of online computer based instruction (CBI). The findings from this study illustrate the effects of dynamic design principles on student learning outcomes.

AN EXPLORATORY STUDY: DYNAMIC DESIGN IN STORYBOARDING TO CREATE ONLINE CBI PROGRAMS

COMPUTER BASED INSTRUCTION

Computer-based instruction (CBI) refers to the use of a computer as an *instructor*. The computer is used to deliver instruction, interact with students, or provide help and resources (Liu & Johnson, 2003). The computer may make use of a program to deliver an instructional unit, tutorial, drill, simulation, instructional game, or test (Alessi & Trollip, 2001). Although advocates claim that using computer based instruction can produce better learning experiences than using traditional delivery methods (Gagne & Medsker, 1996; Harel & Papert, 1991; Jonassen & Henning, 1999; Lebow, 1993; McDaniel & Liu, 1996; Merrill, 1991), research has not backed up these claims. One reason for the lack of research evidence that CBI is more effective than other delivery systems may be that many existing CBI programs are either poorly designed or inappropriate for the learning goals to which they are appliked (Alessi & Trollip, 2001; Liu &Velasquez-Bryant, 2003). To meet specific needs of instruction and produce better learning outcomes, some educators have developed their own CBI programs using instructional authoring tools such as *Authorware, Director, ToolBook or HyperStudio* (Liu & Johnson, 2003).

The development of a CBI program that makes use of the ADDIE model will follow a five-phase developmental life cycle: (a) *Analysis*, (b) *Design*, (c) *Development*, (d) *Implementation*, and (e) *Evaluation* (Beasley, 1999; Burch, 1992; Yourdon, 1988). In the first phase *Analysis*, the designer determines *What to Do*, identifying needs, problems, goals, scope, contents, the main structure, and all the requested tasks of the CBI program. In the second phase, *Design*, the designer determines *How to Do*, designing specific procedures corresponding to all the tasks in the first phase. In the third phase, *Development*, all designs and plans are converted into operations. In the fourth phase, *Implementation*, each task and the entire CBI program are implemented. Finally, *Evaluation* of the program is performed.

STORYBOARDING

One important task in the second phase (*Design*) is storyboarding. Storyboarding involves producing a series of cards that illustrate screen template, functional areas, and all information to be placed on each screen of a CBI program. All the cards together form the storyboard of the CBI program, which serves as the blueprint of the program. In a CBI program, the primary functional areas include title, informational/instructional text, graphics, directions, feedback, icons and navigation options. Screen templates show the exact positions of these functional areas that vary based on the purpose of each screen. Each card in the storyboard also provides descriptive information required to produce the text, graphics, animations, audio, video, and button-links for navigation.

Traditionally, storyboard cards make use of 5 by 7 index cards. One card equates to one screen. Figure 1 shows an example of a screen template on an index card in which functional areas are put into specific positions. For example, the graphic is on the right, text on the left, and buttons at the bottom of the screen. Notice that on the index card, all the information is presented in a single dimension, by written or typed texts only and each card focuses on the state of that single card. The connections to other cards are not visually shown as a procedure. Basically, index-card storyboarding applies the principles of static design.

Some studies suggest that multimedia-based instructions have improved student learning in a variety of areas, such as engineering, physics, medicine, and education (Cadoni, Botturi, & Forni, 2008; Chol, Lee, & Jung, 2008; Stelzer, Gladding, Mestre, & Brookes, 2009). Therefore, educators have started to design their own CBI programs making use of more multimedia tools and applications. However, it has been found that index-card storyboarding is not able to present all the necessary information on each screen or in the entire CBI program when multimedia plays a dominant role. Although a presentation tool in nature, MicroSoft's *PowerPoint*, with its multimedia functions, has been considered a potentially useful multimedia storyboarding tool when designing a multimedia CBI program.

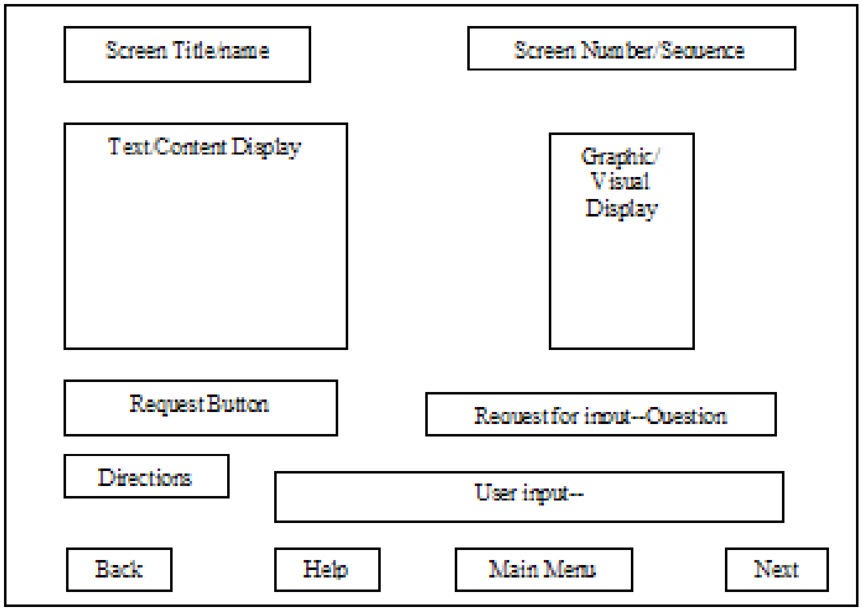


Figure 1. Index Card Storyboard Sample

TYPE II APPLICATION AND DYNAMIC DESIGN FOR STORYBOARDING

Using *PowerPoint* as a storyboarding tool for a CBI design reflects all five features of a Type II application. When a user creates a slide (each slide is analogous to a storyboard card), he/she determines what will be on the screen (Type II application feature 2). The user, rather than the program actively determines what will appear on each slide, (features 1 and 2). The completion of the slides (storyboard cards) for all the screens in the CBI program is definitely a "creative task" (feature 4). *PowerPoint* also allows the user to control the user-machine "interaction" and determine "when to show what" (feature 3). Finally, *PowerPoint* is a highly flexible program with many complex capabilities that will take "many hours" for any developer to learn. These capabilities include functions of such as basic design, creation of media files, coding macros that can control objects on the slides, Web presentation, etc. (feature 5).

When dynamic design is used in *PowerPoint* storyboarding, one or any combination of the four principles of dynamic design can provide strategic guidance for the decisions on each card as well as for the entire storyboard. First, *PowerPoint* storyboarding can be a *nonlinear* process during which the storyboard cards do not have to be completed one after another.

Second, it can be done at *multiple dimensions*: by functional areas on each screen, by graphic, by layout, or by media. For example, the designer can first design the templates for the screens and complete the text contents for all screens; then collect and complete the graphics for all screens and put them on to all the cards; and then develop all media files (video or sounds, for example). Some of the work can also be done simultaneously.

Next, it can focus on the *process* of the whole CBI program rather than each single screen. The decisions are made through the entire CBI program taking into consideration (a) the connections and interactions among screens, (b) the flow of contents, and (c) the analyses of information, audience, and original goals and objectives of the CBI program.

Last, it can be an *open-ended* design as described in the fourth feature of dynamic design. Using *PowerPoint*, the designer can always add or delete slides and adjust the contents of each screen when the program needs to update. The four features of dynamic design are clearly reflected from *PowerPoint* storyboarding. Figure 2 is an example of a storyboard created with *PowerPoint*.

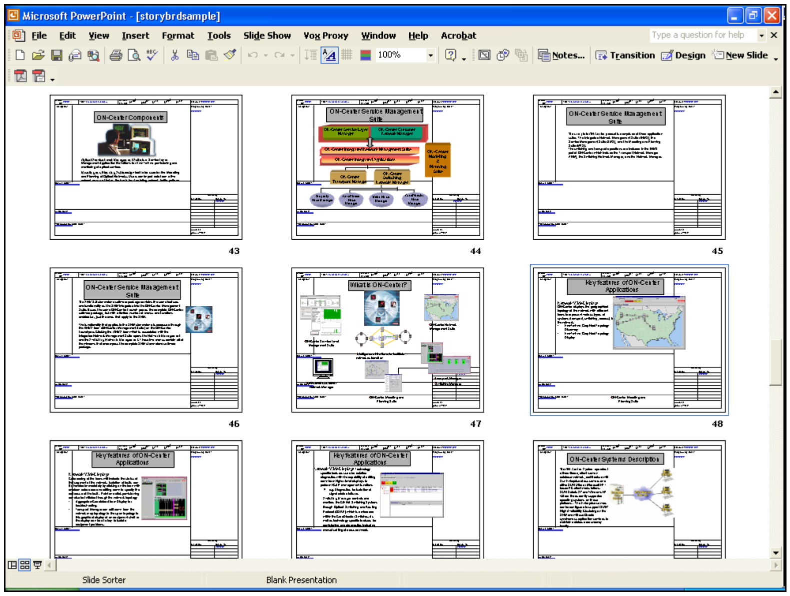


Figure 2. PowerPoint Storyboarding Sample

PURPOSES AND RESEARCH QUESTIONS

PowerPoint appears to have more advantages than index card storyboarding. However, no research has been done to validate this advantage, however. The purpose of the study to be reported in this section was to compare index card storyboarding with *PowerPoint* storyboarding and their effect on quality of design. Four design related qualities of a CBI program were investigated: screen design, interaction, orientation, and navigation.

The research questions examined in this study were:

1. Are there significant mean differences in design quality (as measured by the combination of screen design, interaction, orientation, and navigation) of CBI units created with different storyboarding methods (index card vs. *PowerPoint*)?
2. Are there significant mean differences in *screen design* of CBI units created with different storyboarding methods (index card and *PowerPoint*)?
3. Are there significant mean differences in *interaction design* of CBI units created with different storyboarding methods (index card and *PowerPoint*)?
4. Are there significant mean differences in *orientation design* of CBI units created with different storyboarding methods (index card and *PowerPoint*)?
5. Are there significant mean differences in *navigation design* of CBI units created with different storyboarding methods (index card and *PowerPoint*)?

PARTICIPANTS

The participants in this study were 72 graduate students (56 females and 16 males, ages 28 to 52, Mean = 34.6, *SD* = 1.98), enrolled in six sections of a CBI design course in a state university in the eastern United States. This course was an advanced course in the master’s program in instructional technology at that university. All participants were skilled computer users, but had not had any previous CBI design experiences.

PROCEDURES

A convenience sample was selected from six classes of a design course titled "Theory and Design of Computer Based Instruction." This course required students to learn CBI design theories and tools, complete CBI development procedures, and use an authoring tool (*Director*) to create two multimedia CBI units on selected learning topics. The two CBI units were two equal-weighted projects

In each class, students were randomly assigned into two groups using either Index card (static design methods) or *PowerPoint* (dynamic design methods) as storyboarding tools to create CBI units. Each CBI unit was scored on its design quality. From each class, six students' scores were randomly selected from each group (index card or *PowerPoint*), making a total of 12 students from each class, and a total of 72 (from the six classes) as the sample for the study.

The random grouping was employed for the first project, and quality scores of the first CBI units from this project were used for the data analysis. It is believed that using the data from the first project will strengthen the study by controlling student previous CBI design experience. After the first project, students switched the methods, or they could choose either method for their second project.

INSTRUMENTS AND MEASUREMENTS

The instrument used to evaluate students' multimedia instructional design was a criteria list that has been used in many studies (Ivers & Barron, 1998). This instrument evaluates content, language, screen displays, visual images, interactions, orientation, navigation, input, response analysis and feedback, help, evaluation and record keeping, and technical consideration. Under each item, there was a detailed checklist. Each quality item was scored from 1 to 10, and higher scores indicated better qualities.

We selected four quality items for this study, because the quality of storyboarding directly influences these four design qualities: (a) screen design – the screen frames are properly designed to achieve balance, harmony and simplicity; color, text styles and special effects are used appropriately; (b) interaction – interaction possibilities are maximized and properly designed; (c) orientation – a natural sense of dialogue is created, users can control the pace or sequence, screens were properly labeled so users can easily find out where they are; and (d) navigation – users can easily get where they want to go (see Appendix).

Design quality of each CBI unit was scored by three evaluators. The average scores were used for data analysis. The maximum score for each quality item is 10, and for the total quality score used in this study is 40.

DATA ANALYSES AND RESULTS

Multivariate analysis of variance was conducted in which the dependent variable (DV) was the combined *Design Quality*, as measured by *Screen Design* (DV_1), *Interaction Design* (DV_2), *Orientation Design* (DV_3), and *Navigation Design* (DV_4). The independent variable (IV) was *Storyboarding Method* at two levels: (a) static design methods with Index card, and (b) dynamic design methods with PowerPoint. The results from the data analysis follow.

First, the Box's Test was not significant ($F_{(10, 23426)} = 1.493$, $p = 0.135$), indicating that the assumption of equal variances is not violated. Therefore, the Wilks' Lambda test statistic was used (Mertler & Vannatta, 2002, p. 126). MANOVA results indicate that storyboarding method significantly affected the combined DV of *Design Quality* (Wilks' $\Lambda = 0.088$, $F_{(4, 67)} = 173.192$, $p < 0.0001$, $\eta^2 = 0.912$).

A univariate ANOVA was conducted as the follow-up test. ANOVA results indicate that *Screen Design* significantly differs for storyboarding methods ($F_{(1, 70)} = 170.992$, $p < 0.0001$, $\eta^2 = 0.710$), so does *Interaction Design* ($F_{(1, 70)} = 323.578$, $p < 0.0001$, $\eta^2 = 0.822$), *Orientation Design* ($F_{(1, 70)} = 342.810$, $p < 0.0001$, $\eta^2 = 0.830$), and *Navigation Design* ($F_{(1, 70)} = 608.102$, $p < 0.0001$, $\eta^2 = 0.897$).

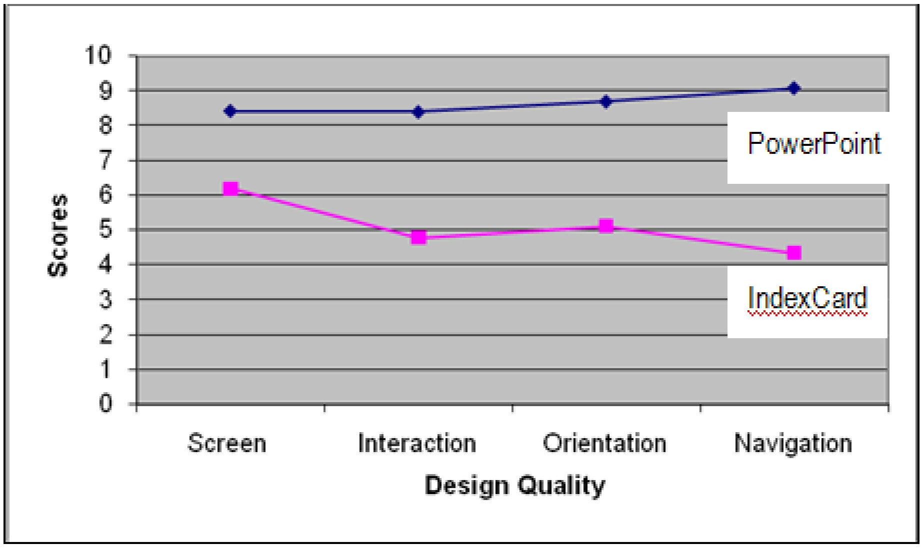


Figure 3. Means of the Design Qualigy

The post hoc results, as well as results from the independent samples *t* tests for the four design quality dependent variables show that CBI units created with PowerPoint storyboarding method had higher scores on all four design variables than those created with the Index Card method (as shown in Figure 3): *Screen Design* (t =13.076, p <0.0001, mean difference = 2.222), *Interaction Design* (t = 17.988, p <0.0001, mean difference = 3.611), *Orientation Design* (t = 18.515, p <0.0001, mean difference = 3.583), and *Navigation Design* (t = 24.660, p <0.0001, mean difference = 4.750). Table 2 shows the mean and standard deviations of quality scores by storyboarding method.

Table 2. Design Quality Scores by Storyboarding Method

	Screen		**Interaction**		**Orientation**		**Navigation**		**Total**	
	M	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**	**M**	**SD**
PPT (*Dynamic*)	8.42	0.73	8.39	0.90	8.69	0.92	9.08	0.99	34.58	2.69
Index Card (*Static*)	6.19	0.71	4.78	0.79	5.11	0.70	4.00	0.58	20.42	2.17

CONCLUSIONS

All five research questions for this study were answered and the results indicate that when using a Type II application (*PowerPoint* as the storyboarding tool), with dynamic design principles, students could produce a better CBI design than when using a Type I application (traditional index-card storyboarding tool), with static design principles. That is, Type II application and dynamic design together contributed to a better quality of student learning.

The next examples will demonstrate some "*how-to*" strategies or methods of using dynamic design with another Type II application in different learning areas or different educational settings.

EXAMPLE ONE: DYNAMIC DESIGN IN USING *LOGO* TO LEARN GEOMETRIC CONCEPTS

This example involves use of a Type II program (the *Logo* computer program). The examples also make use of dynamic design in geometric concept learning. Theoretical foundations, a three-level and two-process learning model, and *Logo* learning tasks developed with dynamic design principles are described.

GEOMETRIC THINKING AND LEARNING

According to van Hiele's theory of geometric thinking, geometric learning occurs only when children's thinking and learning about geometry have advanced through three levels (Fuys, Geddes, & Tishchler, 1988; Liu & Cummings, 1997, 2001; van Hiele, 1986, 1997):

1. *Visual level*, at which children identify and think about the geometric shapes by the visual appearance of the shapes and their similarity to real world objects (van Hiele, 1999);
2. *Descriptive/analytic level*, at which children are able to think about geometric shapes based on the characteristics of the shapes (Liu & Cummings, 2001; van Hiele, 1999); and
3. *Abstract relational level*, at which children are able to formulate an abstract concept of geometric principles by integrating the visual information obtained from level 1 and the understanding of characteristics of shapes gained from level 2 (Liu & Cummings, 1997, 2001; van Hiele, 1999).

In this three-level model, van Hiele explains the *progression* of children's capability in geometric thinking from one level to next. However, the model does not describe the *processes* that must be in place to make the transition occur (Liu, 1999b). Liu and Cummings (1997, 2001) have proposed and examined two thinking processes that advance movement through van Hiele's three-level hierarchy:

1. *Concrete-abstract process* (CA), during which children experience concrete objects, abstract characteristics of the objects/shapes, and formulate geometric concepts and ideas; and
2. *Abstract-concrete process* (AC), during which children apply the newly learned concepts and ideas to solve other similar or more advanced geometric problems.

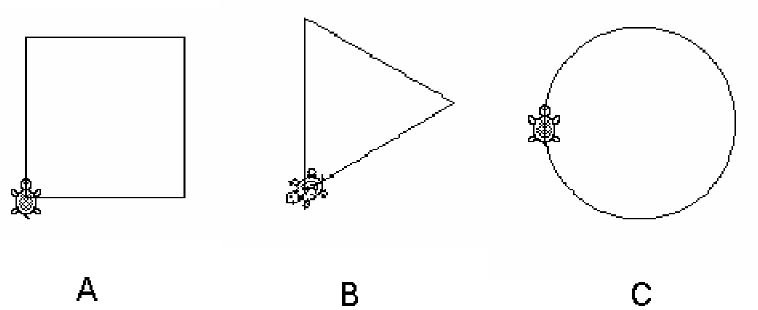
Through concrete-abstract (CA) processes, children's geometric thinking advances from visual level to abstract/relational level. Liu and Cummings (2000) also proposed that "van Hiele's abstract/relational level is not the highest level at which children can think about geometry" (p.88). Once they reach that level, children are able to apply thinking process at an even higher level, abstract-concrete (AC) thinking process, to solve concrete problems (Liu & Cumming, 2000; Liu, 2000). A new model can be produced from combining van Hiele's three levels (1986, 1997) with Liu and Cummings' two processes (1997, 2001) of geometric thinking. With this new model, one of the major purposes of geometry instruction is to activate children's thinking through the CA – AC processes across all three levels.

USING LOGO AS A TYPE II APPLICATION
IN GEOMETRIC THINKING AND LEARNING

Logo is a potential tool for developing this three-level two-process model to stimulate geometric thinking, and improve geometric learning (Liu & Cumming, 2000). *Logo* is a computer language that combines "the capabilities of artificial intelligence with the theories of Jean Piaget" to allow a learner to build his/her own "intellectual structures through estimation, interaction, experience and revision" (Harper, 1989, p.1). With *Logo*, a learner can enter directly into the world of *turtle* geometry, and geometric thinking is possible for any learners without a series of prerequisites (Harper, 1989; Judd, 1983; Maddux & Johnson, 1997).

Apparently, the use of *Logo* is a Type II application. The *Logo* language provides an environment that allows a learner (a) to perform intellectual activities (Type II application feature 1) such as "setting a problem to solve, making choices, playing with the problem, experimenting and trying out solutions" (Harper, 1989, p.1); (b) to control the screen turtle-activities and the interactions (features 2 and 3); and (c) to complete tasks as creative as possible (feature 4). Although a learner can get start on *Logo* very quickly, he/she still needs to spend more hours to explore and learn more functions of *Logo* (feature 5).

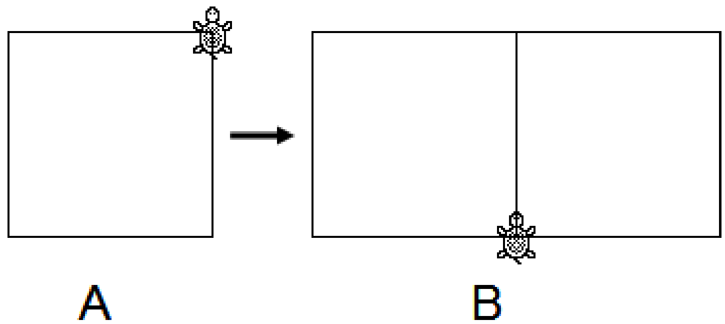
Findings from a previous study (Liu & Cummings, 2001) suggest that geometric thinking/learning occurs when two types of *Logo* learning tasks are performed: (a) code-shape task, and (b) shape-code task. First, in a code-shape task, a learner is given the *Logo* code of a particular shape, for example, "repeat 4 [forward 100 right 90]," and he/she then inputs the code to produce the shape—a square (Figure 4-A). After repeating several codes to produce a set of square shapes of different sizes, the learner should be able to think about the shape of a square based on the characteristics of a square. In this procedure, the learner performs *concrete-abstract* thinking, and his/her thinking and learning about geometry has advanced from *Visual level* to *Descriptive/analytic level.*



A: repeat 4 [forward 100 right 90]
B: repeat 2 [forward 100 right 120] forward 100
C: repeat 360 [forward 1 right 1]

Figure 4. Simple Logo Tasks

Next, the learner should complete a set of simple and comprehensive shape-code tasks in order to progress to the final *Descriptive/analytic level*. To complete a simple shape-code task, the learner needs to create *Logo* codes to produce a simple given shape (for example, a square shape of particular size). In a comprehensive shape-code task, the learner is given a geometric shape that includes the shape learned from the previous code-shape task, for example, a rectangle that consists of two square shapes (as in Figure 5-B), and he/she then needs to create the *Logo* codes (see Figure 5 codes) to produce it. Through this procedure, the learner performs *abstract-concrete* thinking, and his/her thinking and learning about geometry has advanced to the *Abstract relational level*, being able to formulate an abstract concept of geometric principles of a square by integrating the visual information obtained from *Visual level* and the understanding of characteristics of shapes gained from *Descriptive/analytic level*.



A: repeat 4 [left 90 forward 100]
B: repeat 3 [right 90 forward 100]

Figure 5. Comprehensive Logo Tasks

DYNAMIC DESIGN OF LOGO TASKS

Again, the three-level two-process model has provided the guidance to determine procedures and contents of geometric learning tasks with *Logo*, and the strategies and methods to implement the procedures with the learning contents could be determined with either *Static* or *Dynamic* design principles. For example, in the design of *Logo* tasks to learn the three geometric concepts—*square*, *rectangle*, and *triangle* (shown in Figures 4 and 5), dynamic design principles could be applied as in the following.

Multiple Dimensional Learning. In this learning context, the term “dimension” is interpreted as *a different type of task that can be performed alone*. The *Logo* tasks could be set at several dimensions:

1. code-shape exercises, to produce a particular given shape;
2. code-shape exercises, with different values to produce different sizes of the shapes;
3. shape-code exercises, to write the code for a particular given shape;
4. shape-code exercises, to produce the codes for different sizes of the shapes;
5. comprehensive code-shape exercises, to construct a shape with a combination of two or more different shapes; and
6. comprehensive shape-code exercises, to write a set of codes for a shape combined with two or three different shapes.

The first four exercises should be completed first, but they do not have to be performed linearly. The last two comprehensive exercises would have better results if they were done after the first four exercises. The multiple dimensional exercises reflect all the procedures in the three-level two-process model.

Process Oriented Learning. Working through the above six *Logo* activities, a set of process-oriented tasks could be used to improve geometric concept learning:

1. recording the results of each step of a given task into a worksheet, such as the length of side, degree of angles, the shapes, or the set of codes;
2. reviewing the entire learning process with the information input into the worksheet;
3. analyzing and interpreting the connections among the results from different learning tasks; and
4. summarizing the results from all procedures and formulating the concepts.

The process-oriented learning tasks help a learner to learn from a structural view of knowledge, with which the learner is able to complete the required geometric learning and thinking processes.

Open Ended Exploration. Theoretically, open-ended learning tasks would reinforce the abstract-concrete thinking at a higher level. Such learning tasks can be designed in a variety of ways according to the learning objectives. In the present learning context, for example, some open-ended tasks can be:

1. creating a comprehensive problem that applies the knowledge or skills generated from two or more of the six exercises;
2. providing all possible solutions, with a variety ways to solve the problem; and
3. demonstrating the reasoning procedures for each of the solutions.

These *multiple-dimensional, process-oriented,* and *open-ended* learning tasks have been used, examined, and revised by the present authors with a series of case experiences, and in a pilot study. Outcomes from the cases and pilot study have consistently shown positive impacts of dynamic design on geometric learning. An experimental study has been planned to examine the *effectiveness*. The results and findings will be presented in a future article when the experimental study is completed.

MORE EXAMPLES OF DYNAMIC DESIGN IN DIFFERENT AREAS

In the above exploratory study and Example One, dynamic design principles were employed in two entirely different areas: (a) a CBI storyboarding design, and (b) a *Logo* task design. Although specific tasks in the two areas were different, the strategies derived from dynamic design principles were the same. This is the key point: dynamic design principles can be used strategically in many educational applications, but the decisions about specific tasks are based on the theories and content of that particular field. Generally, when a Type II application is used with dynamic design for the purpose of improving learning, four components need to be included:

1. theoretical foundation for the learning or application area
2. objectives, contents, and tasks of the learning or application
3. Type II applications
4. dynamic design strategies

The next three examples will introduce some practical methods to use dynamic design in (a) online course communication, (b) online counseling, and (c) course management. Due to limitation of length of the article, emphases will be on the dynamic design strategies applied in these three areas.

EXAMPLE TWO: DYNAMIC DESIGN IN ONLINE COURSE COMMUNICATIONS

In an online course, online communication is always a critical factor influencing the quality of online teaching and learning (Boer & Collis, 2001; Coombs & Rodd, 2001; Liu, 2003). This example introduces the design of online communications in an online course delivered through *WebCampus*. Online communications are conducted with built-in tools such as course mail, chat, discussion board, or live classrooms. The course system and communication tools can be considered Type II applications as they all conform to the criteria for Type II applications discussed earlier in this article (Liu & Johnson, 2004; Liu & Maddux, 2005). To achieve the learning goals and objectives set for the course, online communications can be designed with dynamic design principles as in the following.

Multiple Dimensional Communications. Online communication activities are usually designed to achieve different goals at several dimensions:

1. To obtain information, get help, or work on team projects, online communications could be conducted at three levels (individual one-to-one level, group many-to-many level, and the class one-to-many or many-to-many level).
2. Online communications could also be performed in different formats or with different media. For example, text-based journals could be used for theme discussion using video conference for team communications, or using live classroom to get help from the instructor.
3. Synchronous communication could be used for immediate feedback from the instructor or classmates, and asynchronous communications could be used for on-going discussions.
4. Communications could be completed with different tools such as course email, discussion board, chat room, or with video conference tools like Skype.

Process-Oriented and *Open-Ended Communications* are asynchronous communications, usually designed for the course content theme discussion. All the class members can participate in the discussion at their convenience, with sufficient time to think and prepare for the discussions.

To integrate the technology tools into the communication activities and achieve the expected learning outcomes, the ADDIE instructional design model should be applied. According to the authors' experiences, those multiple dimensional online communication activities, when well designed, motivate and improve student online learning and reduce the level of online learning anxiety (Liu, 2005, 2007). The activities at each dimension make up multiple-dimensional online communications.

EXAMPLE THREE: DYNAMIC DESIGN IN ONLINE COUNSELING

This example involves the design of learning activities for counseling education students to learn and improve their knowledge and skills of online counseling. Type II technology tools for online counseling include web publishing, synchronous communication tools such as *Skype*, discussion board, and email. Dynamic design principles can be used in the design of online counseling activities.

Non-Linear Learning Tasks. To prepare counseling students to conduct online counseling, they are required to complete learning tasks in three foundation areas:

1. content knowledge and skills, such as counseling theories, models, and skills;
2. technology tools needed to perform online counseling; and
3. design of online counseling, including analyzing, planning, designing, implementting, and evaluating all the online counseling activities.

In each area, there is a list of learning tasks and instructional materials are provided. Tasks in the three areas do not have to be completed in a linear order.

Multiple Dimensional Activities. Online counseling activities are expected to be conducted at several dimensions according to the purposes of each activity. Typical activities in the online counseling include:

1. individual work such as developing and publishing a counseling information website,
2. individual one-to-one online counseling sessions,
3. group many-to-many online counseling sessions,
4. online session video capture for assessment and evaluation, and
5. group research.

Activities at different dimensions also can be completed interactively, which enables students to better understand the connections among the learning activities, as well as the entire process of online counseling.

Process-Oriented Learning is merged into all the learning tasks and online counseling activities, such as documenting the process of individual activities and of the entire project. In addition, the group research project requires students to develop a personalized, technology-based counseling model, which involves a maximum of *Open-Ended Exploration.*

The online counseling course has been taught three times over the past three years. The online counseling experiences have helped students learn not only online counseling skills, but also the dynamic design strategies (Liu & Gentile, 2008).

EXAMPLE FOUR: DYNAMIC DESIGN IN COURSE DESIGN

The last example shows another unique application in course re-design and management. To better organize a general education introductory course and motivate students to learn actively, dynamic design principles are applied in four course design components (Liu & Maddux, 2005).

1. Course assignment format: *Non-linear* portfolio-based assignments are required rather than linear item-based assignment format.
2. Course delivery format: a *multiple dimensional* format of course delivery is used. The course is delivered partially online rather than online only or classroom only delivery methods.
3. Course pace control: non-sequential self pace control is used that employs a *multiple layer* of pace control, including weekly work and overlap of course work for different course projects through the semester. The due dates are set according to the multiple layer timelines rather than in linear timelines.
4. Course product design: the course product usually is a comprehensive project that reflects student learning from the entire course. The structure and content of the course product is designed in *multiple dimensions* since the development of the product follows *non-linear* procedures and the product has an *open-end* that allows further development.

The authors of the present article have conducted a study to examine the dynamic-design featured course over a four-year period and found that dynamic-design features in a course did increase the probability of students' being satisfied with the course and the instructor. Students also were more likely to be motivated to learn from such a course.

The three examples in this section briefly demonstrate the ways dynamic design can be applied in different learning contexts to achieve better learning outcomes. Again, the dynamic design principles are constant, but the uses of the principles are varied in numerous ways based on the application context.

CONCLUSIONS AND DISCUSSIONS

This article has presented an exploratory study and four examples of using dynamic design principles to improve teaching and learning. Several tentative conclusions can be drawn. First, there may not be constant rules or unvarying methods to follow when using dynamic design principles. The decisions made for each step or each task may involve a combination of nonlinear processes and content knowledge from multiple dimensions. Reviewing the dynamic design applications described in this article, one can see that the same design principles were applied in cases involving CBI design, Logo geometric learning, online communication design, online counseling design, and course restructuring design. Content knowledge and skills vary among the five areas and the learning tasks and activities are different. However, the dynamic design principles applied to the five areas are the same: non-linear, multiple-dimensional, process-oriented, and open-ended (Liu & Maddux, 2005).

Second, to effectively use dynamic design in any educational applications, *dynamic thinking* is needed. In the examples described in this article, dynamic thinking is reflected through all the task decisions. Dynamic thinking may be summarized as a type of non-linear, multiple-dimensional, process-oriented, or open-ended thinking. The authors are planning additional research to investigate the topic of dynamic thinking.

Third, use of Type II applications does require a careful design with dynamic design principles. Literature has already suggested that technology has not been used effectively in the field of education to improve teaching and learning because of the lack of sound design principles (Liu, Jones, & Sadera, 2010; Liu & Velasquez-Bryant, 2003). Since dynamic design represents a different approach to design with a thorough practical foundation, it may be highly useful to practitioners.

Finally, this article has reflected some features of dynamic design: (a) it demonstrates a new approach to design with examples at multiple dimensions; (b) from the connections among all the examples from different fields, the strategies and methods of using dynamic design are formulated—a process oriented presentation; and (c) open-ended exploration is provided as dynamic design could be used in a variety of fields. It is the authors' hope that the information presented in this article could be of interest to other educators. It may also stimulate further research to explore the effective use of dynamic design.

REFERENCES

Alessi, S. M., & Trollip, S. R. (2001, 3rd Edition). *Multimedia for learning: Methods and development.* Boston: Allyn and Bacon.

Beasley, R. E. (1999). Interactive multimedia development: Predesign analysis. *Journal of educational Technology Systems*, *27*(1), 23-42.

Boer, W. D., & Collis, B. (2001). Implementation and adaptation experiences with a WWW-based course management system. *Computer in the Schools*, *17*(3/4), 127-146.

Burch, J. G. (1992). *Systems analysis, design, and implementation*. Boston, MA: Boyd & Fraser Publishing.

Cadoni, E., Botturi, L., & Forni, D. (2008). Learning by seeing: The TEMAS multimedia learning objects for civil engineers. *TechTrends*, *52*(5), 17-21.

Chol, I., Lee, S. J., & Jung, J. W. (2008). Designing multimedia case-based instructions accommodating students' diverse learning styles. *Journal of Educational Multimedia and Hypermedia*, *17*(1), 5-25.

Coombs, S. J., & Rodd, J. (2001). Using the Internet to deliver higher education: A cautionary tale about achieving good practice. *Computers in the Schools*, *17*(3/4), 67-90.

Fuys, D., Geddes, D., & Tischler, R. (1988). The van Hiele model of thinking in geometry among adolescents. *Journal for Research in Mathematics Education Monograph Series No.3.* Reston, VA: National Council of Teachers of Mathematics.

Gagne, R. M., & Medsker, K. L. (1996). *The conditions of learning: Training applications*. Fort worth, TX: Harcourt Brace.

Harel, I., & Papert, S. (1991). *Constructionism*. Norwood, NJ: Ablex.

Harper, D. (1989). *Logo: Theory and practice*. Pacific Grove, CA: Brooks Cole Publishing Company.

Ivers, K. S., & Barron, A. E. (1998). *Multimedia projects in education: Designing, producing, and assessing*. Englewood, CO: Libraries Unlimited.

Johnson, L., Maddux, C., & Liu, L. (1997, Eds.). *Using technology in the classroom*. New York: Haworth.

Johnson, L., Maddux, C., & Liu, L. (2000, Eds.). *Integrating information technology into the classroom: Case studies*. New York: Haworth.

Jonassen, D. H., & Henning, P. (1999). Mental models: Knowledge in the head and knowledge in the world. *Educational Technology, 39*(3), 37-42.

Judd, D. H. (1983, March/April). Programming: An experience in creating a useful program or an experience in computer control. *Educational Computer*, pp. 20-21.

Lebow, D. (1993). Constructivist values for instructional systems design: Five principles toward a new mindset. *Educational Technology Research and Development, 41*(3), 4-16.

Liu, L. (1999a). Different approaches of system analysis in multimedia courseware design. In C. Crawford, N. Davis, J. Price, R. Weber & D. A. Willis (Eds.), *Technology and Teacher Education Annual 1999*, pp. 830-835. Charlottesville, VA: AACE.

Liu, L. (1999b). Technology and geometrical concept learning. *M/SET (Mathematics/Science Education and Technology) Annual Proceedings 1999* (pp.281-285). Charlottesville, VA: AACE.

Liu, L. (2000). PCLogo and mathematical thinking processes. In R. Robson (Ed.), *Mathematics/Science Education & Technology* (pp. 271-276). Charlottesville, VA: AACE.

Liu, L. (2001). Data analysis on the components of multimedia courseware design. In S. Carson, & D. Suhr (Eds.), *WUSS Conference (Western Users of SAS Software) Proceedings* (PP. 174-179). San Francisco, CA: WUSS.

Liu, L. (2003). Communication design for online courses: Effects of a multi-layer approach. In G. Marks, and A. Rodssett (Eds.), *E-Learn in Corporate, Government, Healthcare and Higher Education Annual 2003* (pp.1699-1702). Charlottesville, VA: AACE.

Liu, L. (2005). Communication Design and Student Online Learning. In W. Pearman, M. Mallott, E. Oshiro, R. Stiller, E. Flower, T. Gregson, & D. Yang (Eds.), Hawaii *International Conference on Education: Conference Proceedings* (ISSN 1541-5880) (7p.) Honolulu, Hawaii: HICE.

Liu, L. (2007). A Synchronous Communication Tool for Online Learning: Strategies, Issues and Outcomes. In C. Crawford, R. Carlsen, I. Gibson, K. McFerrin, J. Price, R. Weber & D. A. Willis (Eds.), *Technology & Teacher Education Annual 2007* (pp. 2399-2404). Charlottesville, VA: AACE.

Liu, L., & Cummings, R. (2001). A learning model that stimulates geometric thinking through use of PCLogo and Geometer's Sketchpad. *Computers in the Schools, 17*(1-2), 85-104.

Liu, L., & Cummings, R. (1997). Logo and geometrical thinking: Concrete-abstract thinking and abstract-concrete thinking. *Computers in the Schools, 14*(1-2), 95-110.

Liu, L., & Gentile, T. (2008). Video applications for online counseling: Design and practice. In C. McNaught & C. Montgomerie (Eds.), *Proceedings of ED-MEDIA, Educational Multimedia, Hypermedia and Telecommunications* (pp. 5662-5667). Charlottesville, VA: AACE.

Liu, L., & Johnson, L. (2003). Structure-based versus function-based method in multimedia courseware design. In C. Crawford, N. Davis, J. Price, R. Weber & D. A. Willis (Eds.), *Technology & Teacher Education Annual 2003* (pp. 744-747). Charlottesville, VA: AACE.

Liu, L., & Johnson, L. (2004). Static and dynamic design in online course development. In C. Crawford, N. Davis, J. Price, R. Weber & D. A. Willis (Eds.), *Technology & Teacher Education Annual 2004* (pp. 2946-2952). Charlottesville, VA: AACE.

Liu, L., & Jones, P. (2008). Create Web-based multimedia learning applications: Ideas for Web 2.0 and e-learning 2.0. In C. McNaught & C. Montgomerie (Eds.), *Proceedings of ED-MEDIA, Educational Multimedia, Hypermedia and Telecommunications*. (pp. 4601-3606). Charlottesville, VA: AACE.

Liu, L., Jones, P., & Sadera, W. (2010). An investigation on experienced teachers' knowledge and perceptions of instructional theories and practices. *Computers in the Schools, 27(1)*. 20-34.

Liu, L., & Maddux, C. (2005). Influences of course design on student evaluations: An initial logistic prediction model. *Journal of Excellence in College Teaching, 16*(1), 125-148.

Liu, L., Maddux, C., & Johnson, L. (2008). Assessment of Integration of Technology in Education: Countering the "No Significant Differences" Argument. *Computers in the Schools, 25*(1/2), 1-9.

Liu, L., & Velasquez-Bryant, J. N. (2003). An information technology integration system and its life cycle: What is missing? *Computers in the Schools, 20*(1-2), 93-106.

Maddux, C. D. (2003). Twenty years of research in information technology in education: Assessing our progress. *Computers in the Schools, 20*(1/2), 35-48.

Maddux, C. D., & Johnson, D. L. (1997). Logo: A retrospective. *Computers in the Schools, 14*(1/2), 1-8.

Maddux, C. D., & Johonson, D. L. (2006). Type II applications of information technology in education: The next revolution. *Computers in the Schools, 23*(1/2), 1-5.

Maddux, C. D., Johnson, D. L., & Willis, J. W. (2001). *Educational computing: Learning with tomorrow's technologies*, 3rd Ed. Boston, MA: Allyn and Bacon.

McDaniel, K., & Liu, M. (1996). *A study of project management techniques for developing interactive multimedia programs: A practitioner's perspective*. ERIC Document Reproduction Service No. ED 396 712.

Merrill, M. D. (1991). Constructivism and instructional design. *Educational Technology, 31*(5), 45-53.

Mertler, C. A., & Vannatta, R. A. (2002). *Advanced and multivariate statistical methods: Practical application and interpretation* (2nd ed.). Los Angeles, CA: Pyrczak Publishing.

Parr, J. M. (2002). *A review of the literature on computer-assisted learning, particularly integrated learning systems, and outcomes with respect to literacy and numeracy: A report to the New Zealand Ministry of Education*. Auckland, New Zealand: The University of Auckland.

Reiser, R. A., & Dempsey, P. L., & T. J. (2007). *Trends and issues in instructional design and technology*. Upper Saddle River, NJ: Pearson/Merrill Prentice Hall.

Smith, P. L., & Ragan, T. J. (2005). *Instructional Design*. Danvers, MA: Wiley.

Stelzer, T., Gladding, G., Mestre, J. P., & Brookes, D. T. (2009). Comparing the efficacy of multimedia modules with traditional textbooks for learning introductory physics content. *American Journal of Physics Education*, *77*(2), 184-190.

Van Hiele, P. M. (1986). *Structure and insight*. Orlando: Academic Press.

Van Hiele, P. M. (1997). *Structure*. Zutphen, Netherlands: Thieme.

Van Hiele, P. M. (1999). Developing geometric thinking through activities that begin with play. *Teaching Children Mathematics*, *5*(6), 310-317.

Yourdon, E. (1988). *Managing the system life cycle*. Englewood Cliffs, NJ: Yourdon Press.

APPENDIX

CBI DESIGN QUALITY CHECKLIST

1	***Screen Design***	The frames are properly designed to achieve balance, harmony and simplicity.	
		Color and text font styles are used appropriately.	
		Animation, video, special effects are used properly.	
		The important ideas, concepts, terms or procedures are highlighted.	
		Purpose of the screen is clearly focused.	
2	***Interaction***	Instructions on how to interact is provided.	
		Interactions are designed according to the purpose of the function	
		Interactions are designed with a variety of format	
		Feedbacks are provided with each interaction	
		Interaction possibilities are maximized (every 15 seconds).	
3	***Orientation***	Instruction on how to use the program is provided	
		Table of contents is provided	
		A natural sense of dialogue is created.	
		Screens are properly labeled so users know where they are (user orientation).	
		Directions and questions are plainly stated. (adequate instructions)	
4	***Navigation***	Users can control the pace. (How fast or slow to go)	
		Users can control the sequence. (As appropriate to the content.)	
		Users can easily get where they want to go. (user navigation, including Exit)	
		Choices are available as much as possible for those wanting them.	
		Program map is provided.	