

**A CAI SYSTEM FOR LEARNING NATURAL
RESOURCE KNOWLEDGE (NRKS)**

By

XIA CHEN

Bachelor of Science

Zhejiang University

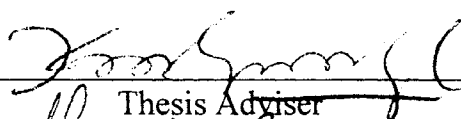
Hangzhou, China

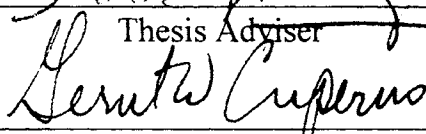
1989

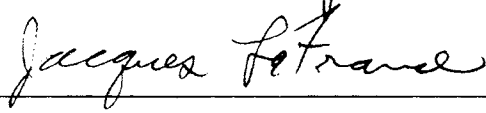
Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 1994

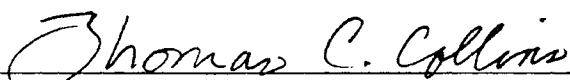
**A CAI SYSTEM FOR LEARNING NATURAL
RESOURCE KNOWLEDGE (NRKS)**

Thesis Approved:



Thesis Adviser






Dean of the Graduate College

ACKNOWLEDGMENTS

I wish to express my sincere appreciation to my advisor, Dr. K.M. George, for his valuable guidance and advice. He gives me much more knowledge that I can not learn from the book. With his encouragement, this thesis has been accomplished. His nice attitude and patience toward the student will be in my mind forever.

I also would like to express my gratitude to other committee members, Dr. J. LaFrance, and Dr. G.W. Cuperus for their helpful suggestions and patient instructions. Special thanks is due to Dr. G.W. Cuperus for his financial support and emotional encouragement while I worked as a research assistant under his supervision. I am grateful to Dr. S. Anderson, Dr. C. Green, Forestry department, and B.J. Chambers, Oklahoma cooperative extension service 4-H program, for their suggestions and expertise. I also like to acknowledge Dr. D. French, Zoology department, for his kindness to let me use his advanced computer hardware and software. In addition, I wish to express my appreciation to all faculties and staffs in the departments of Computer Science and Entomology for their suggestion and friendship.

Finally, my deepest love is expressed to my husband, Dick Q. Cai for his encouragement and moral support throughout my graduate studies.

TABLE OF CONTENTS

Chapter	Page
I INTRODUCTION	1
1.1 Background	1
1.2 Computer-Assisted Instruction (CAI).....	2
1.2.1 CAI History	2
1.2.2 A Design Model for CAI	2
1.2.2.1 Analysis Phase	5
1.2.2.2 Design Phase	8
1.2.2.3 Development and Implementation Phase.....	12
1.3 Hypermedia	12
1.3.1 General View	12
1.3.2 Hypertext	13
1.3.3 Audio in CAI Software	14
1.3.4 Visual Elements in CAI Software.....	14
1.3.4.1 Dynamic Visual Displays.....	15
II SOFTWARE DESIGN	16
2.1 Organization of Quizzes.....	16
2.2 Software Design via Knowledge Pro.....	29
2.2.1 Knowledge Pro Environment.....	29
2.2.2 Software Architecture of NRKS.....	29
2.3 Screen Design	32
2.3.1 Importance of Development Environment	32
2.3.2 Method to Display Graphics	34
2.4 Audio Application Design.....	35
2.5 Dynamic Visual Displays Design.....	36
2.6 User Interface Design	38
III SOFTWARE EVALUATION	40
3.1 Necessity of Evaluation.....	40
3.2 Components of CAI Evaluation.....	41
3.2.1 Evaluating Instructional Design.....	41

Chapter	Page
3.2.2 Evaluating Program Design.....	43
3.2.3 Evaluating Management Design.....	44
IV DISCUSSION AND FUTURE WORK	47
BIBLIOGRAPHY	49

LIST OF TABLES

Table	Page
1 General Steps Found in Instructional Design Systems.....	3
2 Information Related to the Board #1 in Q3-8.....	10
3 Information Related to the Board #2 in Q3-8.....	11
4 Code Segment for Picture Crown Topic.....	34
5 Code Segment for Sound Application Topic.....	36
6 Code Segment for Video Application Topic.....	37
7 Some Segment from Raw Data.....	44
8 An Example of Individual Report.....	45
9 An Example of Class Report.....	45

LIST OF FIGURES

Figure	Page
1 A Three-Phased CAI Design Model [Jannafin, 88].....	4
2 A Frame work For Analysis Phase [Steinberg, 91a].....	5
3 Board Number 1 of 8 in Question 3 (Treefact Quiz).....	9
4 Board Number 2 of 8 in Question 3 (Treefact Quiz).....	10
5 Control Flow Chart for NRKS.....	17
6 Memu Organization (From System Point of View).....	18
7 Memu Window (From User Point of View).....	19
8 Question Window and Hypertext Window (System Point of Wiew).....	20
9 A Sample of Question & Hypertext Window (User Point of View).....	21
10 Hypertext (System Point of View).....	22
11 Fact, Hypertext, and Feedback (System Point of View).....	22
12 Hypertext for Tree Ring Picture (Hyper-Region on Year 1).....	23
13 Hypertext for Tree Ring Picture (Hyper-Region on Year 53).....	24
14 Hypertext for Growth of Ring.....	25
15 Hypertext for More Information about Rings.....	26
16 Fact, Hypertext, and Feedback Windows (User Point of View).....	27
17 Score Reporting (System Point of View).....	28
18 Score Reporting Window (User Point of View).....	28
19 Top Level Structure of NRKS	30
20 The Structure of Treefacts	30
21 The Structure of Question in Treefacts.....	31

Figure		Page
22	The Structure of Children Topics of Hypertext.....	31
23	Screen Layout in Resolution 640×480.....	32
24	Screen Layout in Resolution 800×600.....	33

CHAPTER I

INTRODUCTION

1.1. Background

Learning via CAI has increased dramatically in both academic and nonacademic areas. A survey [Becker, 89] of literature reveals that just in primary education, the average number of computers used in elementary schools has more than tripled between 1985 and 1989, with approximately half of those computers used for CAI. The total number of computers and the CAI lessons used in education increase every year. The increase in demand requires more and more good CAI software.

This thesis studies computer-assisted instruction (CAI), hypermedia, and software evaluation. It develops a CAI software package named Natural Resource Knowledge System (NRKS) using hypermedia as a tool via the expert system language Knowledge Pro. NRKS provides CAI lessons, which consist of five parts namely forestry, urban, wildlife, tree facts, and tree products. Each part emphasizes the knowledge of that particular field. In NRKS we combine text, audio, graphics and animation to help students to study forestry and related subjects. A three-phased design model is applied to NRKS during the developing process.

1.2. Computer-Assisted Instruction (CAI)

1.2.1. CAI History

Computer-assisted instruction first emerged in the late 1950s and since then has developed through four distinct phases in its application to education according to Richard and Luis [Venezky & Osin, 91]. The four phases are: 1) the Age of Engineers (1958-1962), this age was characterized by hardware issues, most products connected input-output devices (such as terminals, slide projectors) to computers for practical interaction. 2) the Age of Acronyms (1962-1967), this age was characterized by their computational and psychological orientations and by their clever acronyms of the major projects (such as CLASS, PLATO, SOCRATES, SAID). 3) the Age of the Titans (1968-1977), in this age, the CAI evaluation study of Murphy & Appel, and Alderman provided a theory about the rational role of CAI in education, the major products of this age were PLATO and TICCIT. 4) the Age of Small Wonders (1977-present), is characterized by the positive attitudes in school toward CAI and by the industrial interest in producing CAI hardware and software. The most significant difference between each phase is its attitudes toward hardware and software need.

1.2.2. A Design Model for CAI

The charge is to produce an effective and efficient CAI software for a variety of learners, goals, environments, and media. To achieve this goal researchers in the field of instructional systems design have developed several models to lead the CAI designer through necessary procedures [Jannafin, 88]. There are slight differences existing among those models. But the general steps in instructional design systems are similar. These steps are listed in Table 1 [Jannafin, 88].

TABLE I
GENERAL STEPS FOUND IN INSTRUCTIONAL DESIGN SYSTEMS

-
1. analyze instructional goals and objectives
 2. decompose goals into tasks and sub tasks
 3. determine the prerequisite skills and the knowledge to be presented in the lesson
 4. determine the constraints (environment, available hardware, time, budget, etc.)
 5. generate instructional alternatives, and make the selection from these alternatives
 6. determine task sequence and develop instruction
 7. program coding, testing, and debugging
 8. evaluate the CAI lesson
 9. modification
-

Most models combine these steps into phases, the methods to combine differs from model to model. The model presented in Figure 1 combines the steps into three phases: *analyze*, *design*, and *develop and implement*. These three phases are connected by evaluation and modification.

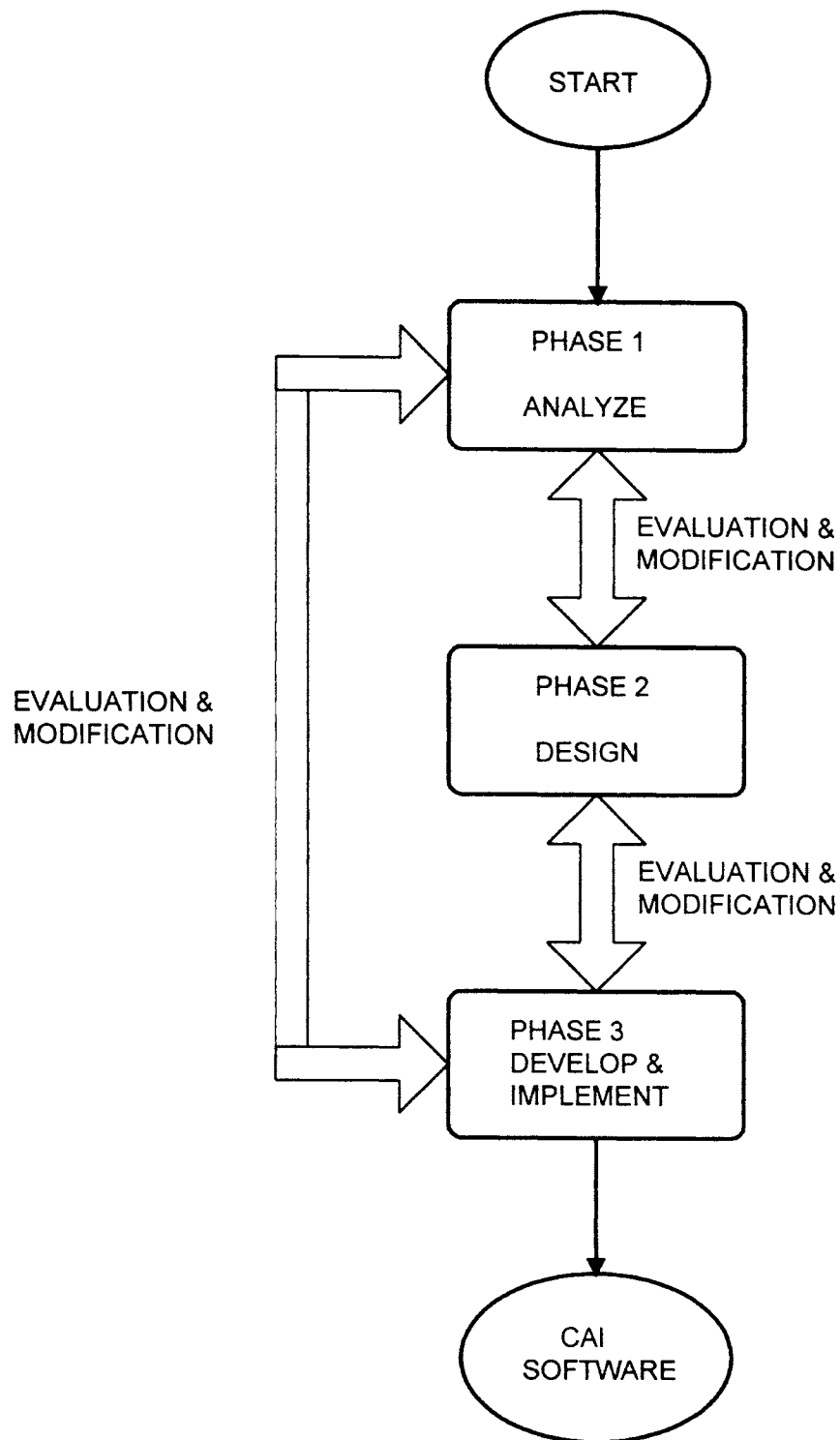


Figure 1. A Three-Phased CAI Design Model [Jannafin, 88]

1.2.2.1. Analysis Phase

This section reviews and relates to NRKS a six-components framework during the first phase [Steinberg, 91a]. These six components are: Target Population, Goals, Task, Instruction, Computer Application, and constraints (see Figure 2.). They interact and this interaction strongly influences the effectiveness of instruction and the success of the CAI software. The discussion of each component and their interpretation in NRKS software are given below.

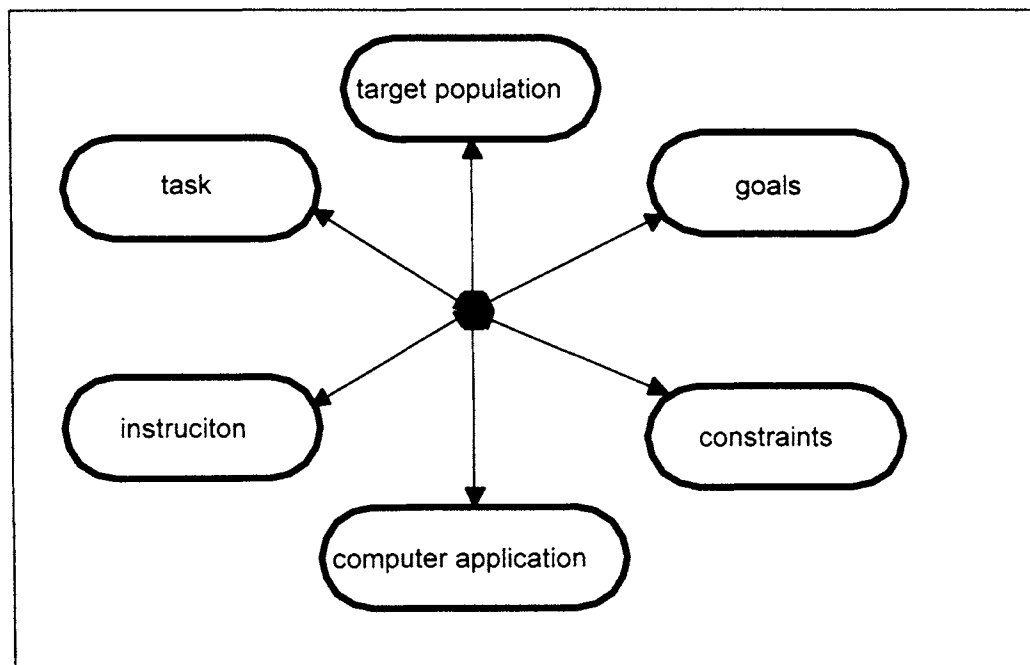


Figure 2. A Framework For Analysis Phase [Steinberg, 91a]

Target Population

Before developing a CAI lesson, the designer has to understand the target audience. Just like all text books have intended readers, so do CAI lessons. The intended

learners (such as third graders) are defined as the target population. A successful CAI lesson should match the characteristics of the target population. Research done by Hirschbuhl [Hirschbuhl, 89] and Roblyer [Roblyer, 88] pointed out that a major cause of unsuccessful CAI lessons in both academic and nonacademic areas is the inappropriate instruction for the intended learners. In NRKS, the target population is 4th or 5th grade students in elementary school, and must be at that target level.

Goals

Lessons usually have a definite purpose, so do CAI lessons. After taking the CAI lesson, the learners should achieve certain outcomes. Steinberg [Steinberg, 91] defined these outcomes as the goals. The goal of NRKS is to help students acquire natural resource knowledge. The knowledge include forestry knowledge (with focus on Oklahoma forestry, and Urban forestry), wildlife knowledge, tree facts knowledge, and the knowledge of tree products. NRKS provides a realistic setting (combination of text, sound, graphics, and animation) to make it more pleasant and more motivating for the students to achieve the goal.

Task

A good strategy to solve problems is to decompose the goal into tasks and sub tasks. As Steinberg [Steinberg, 91] pointed out: "For any given subject, the skills needed to accomplish a task vary with its conceptual difficulty and the cognitive processes required". In NRKS, the task is for students to finish all the quizzes about natural resource knowledge, and the sub task is to answer the questions contained in the quiz. So, the students should have certain reading comprehension, some specific knowledge, and the ability to operate the computer. In order to help students to accomplish the task in NRKS, we check the reading level of text appearing in NRKS by using available software to make sure that 4th and 5th graders can understand them, for some difficult concepts,

we give extra instructions (in the form of text, sound, or graphic). For example, in tree facts quiz question 4, the question is: What is the process by which leaves use the sun's energy to make food? The possible answers are a) phototropism b) respiration c) photosynthesis d) biosynthesis. For most 4th or 5th graders, these answers are difficult to understand, so we give necessary explanation (either in text or combination of text and image) to help the students understand these four terms. We also give specific instructions to students on how to operate the computer in every step.

Instruction

Instruction is defined as planned learning process. There are significant differences (the difference of communication methods, environmental differences etc.) between CAI and traditional instruction. If the developers of CAI lessons are aware of these differences, it will enable their software more acceptable by the students and the instructors. In NRKS, we consider the differences when we make decisions about the nature of feedback, the amount of instruction, and sequencing. NRKS provides multiple instructional paths, according to each student's need, this means that the reader also have certain control of the CAI lesson. For example, in question 4 of tree facts quiz, if the student already knows the terminology, the student can skip the explanation instructions and answer the question directly. Or, they can repeat the explanation any time as needed without worrying about making computer “teacher” bored. The students also can choose the subject from the quizzes according to their interest.

Computer Application

Pritchard [Pritchard, Micceri, & Barrett, 89] pointed out that Appropriate use of computers for instruction involves more than a transfer from traditional instruction with a few ‘bells and whistles’ for motivation. A computer has incredible speed of retrieval of information, has the ability to provide all kinds of sound (such as speech, music, and

nature sound), it also can serve as a projector and a TV set to get either still or movable graphics. What has been mentioned here is only a few of all advantages of a computer. CAI can take several of the advantages of a computer to accomplish some educational processes, even those processes that are difficult for traditional instructions to achieve. In NRKS, we take advantage of the power of a computer, by employing state of the art technology in multimedia.

Constraints

Constraints include hardware, environment, physical conditions, play an important role in the success of CAI. Since 1980s computers and their accompanying technology have become more sophisticated. Now some computer systems include high capacity hard disks and RAMs, CD-ROM driver, audio driver, video driver, and high resolution monitor. But not every user has these equipment available. NRKS considers these different elements, for those users without sound card, we provide a package without sound application, and for those who can not install window applications, we provide a package for DOS.

1.2.2.2. Design Phase

After completing analysis phase, the designer already makes clear all the objectives (task and sub task). In the design phase, the first step is to determine the sequence to meet the objectives. Then, the designer considers several potential solutions, from which the best one is selected based on the information gained from analysis phase and on success likelihood.

We adopted the storyboard concept [Jannafin, 88] to describe the process of achieving the objectives. Storyboards are illustrations to describe every screen change and the important information necessary to the designers, reviewers, and programmers. A storyboard contains a desired screen layout and the information about this screen such

as objective number, board number, previous storyboard(s), possible advance action(s).

Two example storyboards are given bellow:

Example 1

The storyboard is shown in Figure 3 and the related information is given in table II

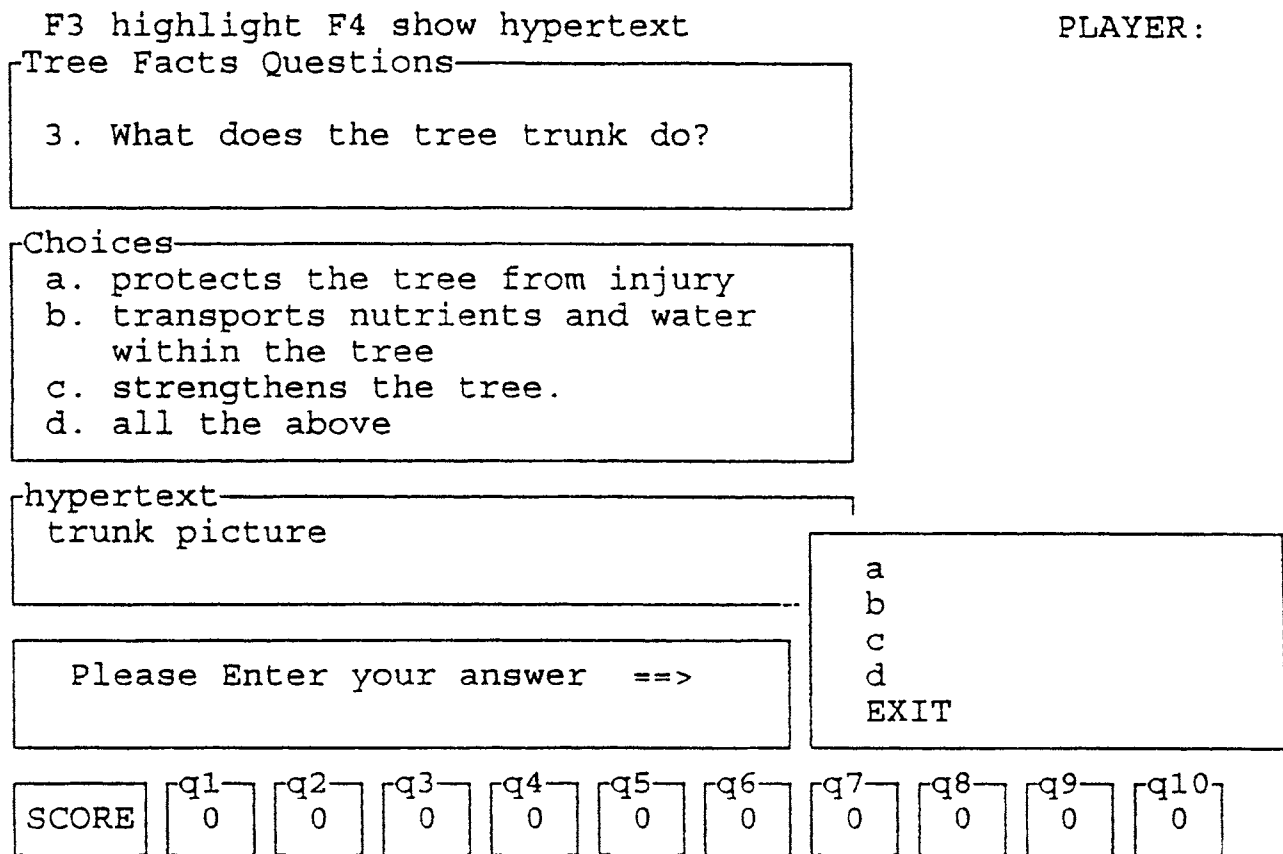


Figure 3. Board Number 1 of 8 in Question 3 (Treefact Quiz)

TABLE II
INFORMATION RELATED TO THE BOARD #1 IN Q3-8

Objective number: *treefact quiz question 3 (Q3)*

Board number: *#1 in Q3-8 (total 8 boards in question 3)*

Previous storyboard: *#5 in Q2-5 (the 5th board in question 2)*

Possible next storyboard: *#2 in Q3-8 or #8 in Q3-8*

Possible advance action(s): *1. press F4 2. move allow sign to select one of the four choices, and press enter*

Comments: *Storyboard #2 in Q3-8 is lead by action 1, and storyboard #8 in Q3-8 will be brought out by action 2*

Example 2

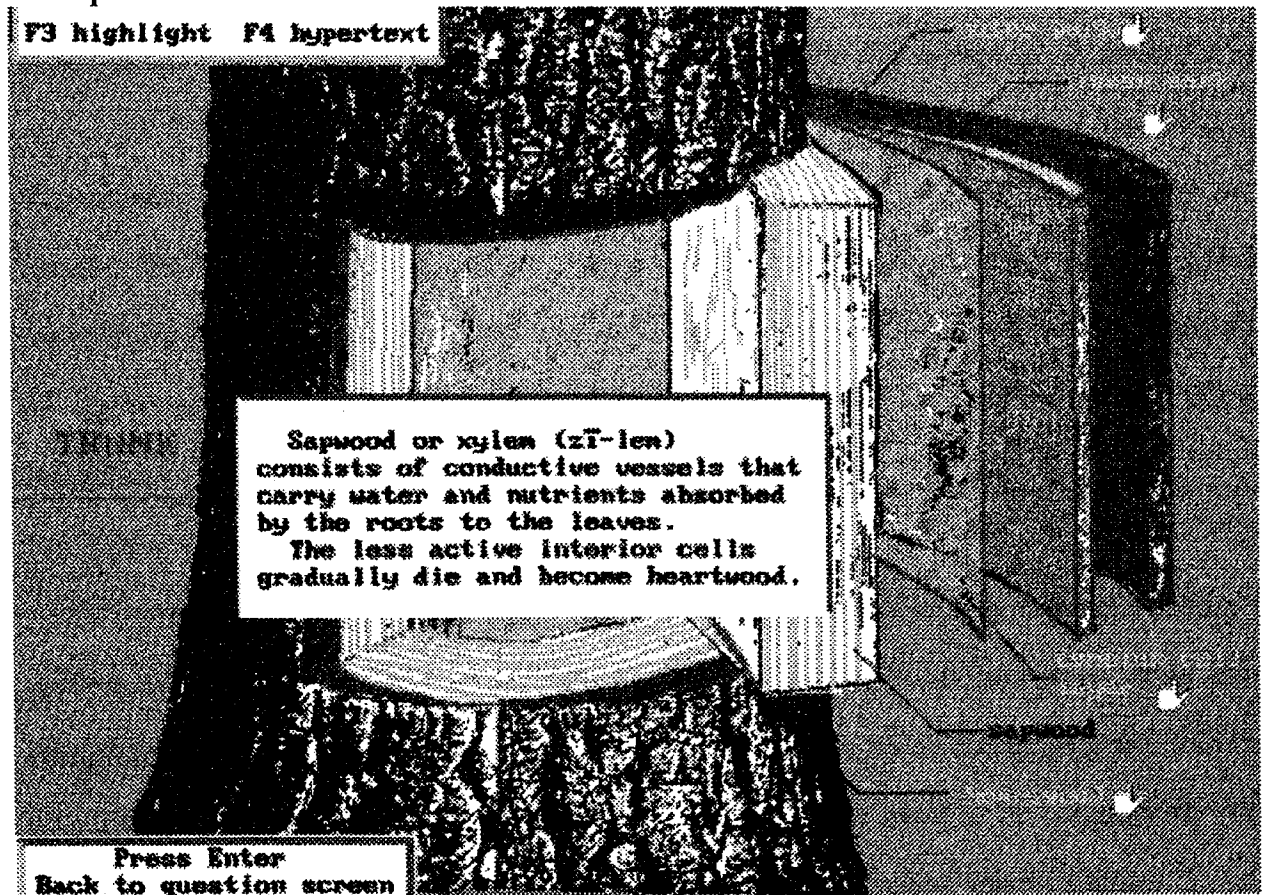


Figure 4. Board Number 2 of 8 in Question 3 (Treefact Quiz)

This example illustrates how graphics and hypertext are used with storyboard. The storyboard is shown in Figure 4. Table III describes the information based on which the board is constructed.

TABLE III
INFORMATION RELATED TO THE BOARD #2 IN Q3-8

Objective number: <i>tree fact quiz question 3 (Q3)</i>
Board number: <i>#2 in Q3-8</i>
previous storyboard(s): <i>#1 in Q3-8</i>
Possible next storyboard(s): <i>#3 in Q3-8 , #4 in Q3-8, #5 in Q3-8, #6 in Q3-8, #7 in Q3-8</i>
Possible advance action(s): <i>1. press F3 highlight one of the five hypertext 2. press F4 to bring out the next screen to illustrate the selected hypertext</i>
Comments: <i>which storyboard will appear next depends on the choice made by the reader. The reader has the opportunity to reenter storyboard #2 of Q3-8 to #8 of Q3-8 anytime from any point within them. But we do not allow the reader reach storyboard #1 of Q3-8 again, because the reader may change their answer after gaining more information from subsequent screens thus losing the accuracy of the quiz.</i>

After all tasks and sub tasks have been depicted by storyboards, the status is reviewed in evaluation and modification process, just as what has been done after completing analysis phase. But the difference is that at this point, the quality and validity of activities from both analysis phase and the design phase need to be considered. The direction may lead to phase 1 or phase 2 depending on the result of the evaluation

process. Also it is possible to proceed to phase 3 to start programming if the first two phases are satisfactory.

1.2.2.3. Development and Implementation Phase

Phase 1 and phase 2 are initial outlines of the lesson. Transfer of these outlines to the CAI lesson will allow transfer to phase 3. In this phase the programmer needs to develop flowcharts, select authoring language, code the program, debug the program, and make the program error-free. The detailed development process of NRKS will be given in chapter II which addresses software design.

As shown in Figure 1, the evaluation and modification process follows the completion of phase 3 also. We will address this process in chapter III. In the next section we review hypermedia as related to NRKS.

1.3. Hypermedia

1.3.1. General View

Hypermedia has become a focus of research and development in educational technology [Kumar, 94]. Hypermedia systems involves students interacting with the computer in the learning process, and provides feedback to the students. Hypermedia technology can revolutionize a student's view of learning [Wishnietsky, 92]. Hypermedia can be defined as organized access to text (words and numbers), aural (sound effects, music, and speech), and visual (still images, video, and animation) elements synthesized into a single integrated presentation system controlled by a computer [Lamb, 92]. NRKS involves all these three elements: text, aural, and visual. Especially in text form, we adopted a technique called hypertext. In the following three sections we will discuss the application of these three elements in NRKS.

1.3.2. Hypertext

In 1965 a student at Harvard University, Ted Nelson began work on a system that would logically link the materials from one class with materials from other classes. Nelson called that system "hypertext", a term defined as non-sequential writing. Nelson conceived of a computer system in which text would be displayed on an interactive screen that provided readers the option of branching to other appropriate text. In 1968, a hypertext system created by Nelson and Andris van Dam was used to link documents for the Apollo moon missions. Another hypertext system, developed in 1968 by Douglas Engelbart, was used at Stanford Research Institute to link computer files [Wishnietsky, 92].

The power of hypertext can be illustrated by "looking up the electronic dictionary" on a computer screen. If the reader requests the electronic dictionary for a particular item, in a short time the related contents will fill the screen. Actually, it provides an Integrated Learning Environment. After selecting the desired section, the designated text appears on the screen [Wishnietsky, 92]. There is a logical pathway between the desired section and the designated text. We can illustrate this by an example. In NRKS tree fact quiz question 1, the student encounters the word "pith", if he or she is not familiar with it, he or she can use a mouse click on this word to bring out a screen without leaving NRKS. The screen says: *The pith is the center most section of a stem or twig.*

For hypertext, the pathway leads to text, and for hypermedia the pathway leads to video and sound in addition to text. These two are comparable. NRKS involves both hypertext and hypermedia.

1.3.3. Audio in CAI Software

Audio in CAI software for second language learning is common. But in other CAI software it is seldom seen, even in those software which did employ audio, most of them just use simple beeps to point out to the user that the wrong button had been pressed

or the wrong answer had been chosen. Only recently more designers of CAI software have become interested in the use of audio.

Aarntzen [Aarntzen, 93] states that: "Audio can *direct attention* to the most important parts of the screen, *draw and maintain the attention* of the learner, also *create a mood to motivate* the learner."

What kinds of roles can audio play in CAI software? Brewer [Brewer, 86] used a list of verbs to describe some fundamental audio functions that could play a part in interactive audio applications. These verbs are: (1) greet, welcome; (2) explain; (3) prompt, request, demand; (4) remind, alert, warn; (5) help, assist, hint; (6) reinforce, praise, flatter; (7) encourage, reassure; (8) advise, suggest, guide; (9) lament, regret; (10) confuse, divert; (11) scare; (12) excite, arouse.

NRKS uses audio to tell a student that his/her answer is correct or incorrect, if it is correct, the computer will play a short piece of music as a reward to motivate the student. Also in the beginning of the quiz, the students will hear greeting music.

Taylor [Taylor, 88] noted that music can be used to: " a) create atmosphere and mood, tension and relaxation; b) underscore the mood; c) change the pace of the program and maintain arousal; d) introduce a new phase of the story; e) unify a series of ideas; f) separate and emphasize the points to be made."

1.3.4. Visual Elements in CAI Software

Graphics play a very important role in people's cognitive process. Studies of visual perception indicate that "a picture is worth a thousand word". House [House, 82] said: "Graphics have been used for centuries to effectively communicate information among people and aid comprehension of complex information. Because patterns and shapes are inherently less abstract than numbers and languages, pictures convey information more readily and permit easier retention than textual or verbal representations of the same information". So, more and more software applications use graphics.

NRKS employs pictorial representation (most of them combine additional text at the same screen, some of them even with audio at same time) for every quiz. On the average there are five pictures for every question.

1.3.4.1. Dynamic Visual Displays

Any type of pictorial or graphical movement during instruction is defined as dynamic visual display (DVD). Dynamic visual displays, including animation and interactive video, have become a primary component of media-based instruction. Research on the instructional roles of DVDs shows that DVDs can be effectively used to facilitate human cognitive process of visually representing imaginable events, actions, and ideas. They can make complex cognitive tasks more concrete and easy to understand by providing motion and trajectory attributes [Park, 94].

Instructional roles of DVDs are identified as (1) *An attention guide* (2) *An illustration aid* (3) *A knowledge representation means* (4) *A facilitator of mental model formation* (5) *A visual analogy or reasoning anchor for understanding abstract and symbolic concepts or processes.*

NRKS involves two kinds of DVDs. One is animation which is implemented in DOS environment, the other one is video which is played in MS Windows.

CHAPTER II

SOFTWARE DESIGN

2.1. Organization of Quizzes

NRKS is organized based on five quizzes. There are a total of fifty questions. Each question has three or more hypertext, and each hypertext employs the combination of text and graphic forms. NRKS uses several different media. So, the organization is a complex task. NRKS quizzes are organized to facilitate the control flow chart shown in Figure 5. The approach taken for NRKS Windows software design is different from the approaches found in the literature. First we design a flow chart of activities (Figure 5). Then we design the software to provide a corresponding user view. Each part of the flow chart is given two interpretations. One interpretation is to facilitate user view and the other is to facilitate system view.

The meaning of the parts of the flow chart in figure 5 is quite different from system point of view and from user point of view. In this section a detailed discussion is provided emphasizing on these two views. Each part of the flow chart will be examined from both points of view. For the user point of view, the screen as seen by the user is provided. For the system point of view, software components and their interactions will be illustrated. As show in Figure 5, the function of parts ① and ② is to choose one of the five quizzes. In the window Version of NRKS, we use a menu to implement this function. The two views of menu setting are shown in Figure 6 and Figure 7 respectively.

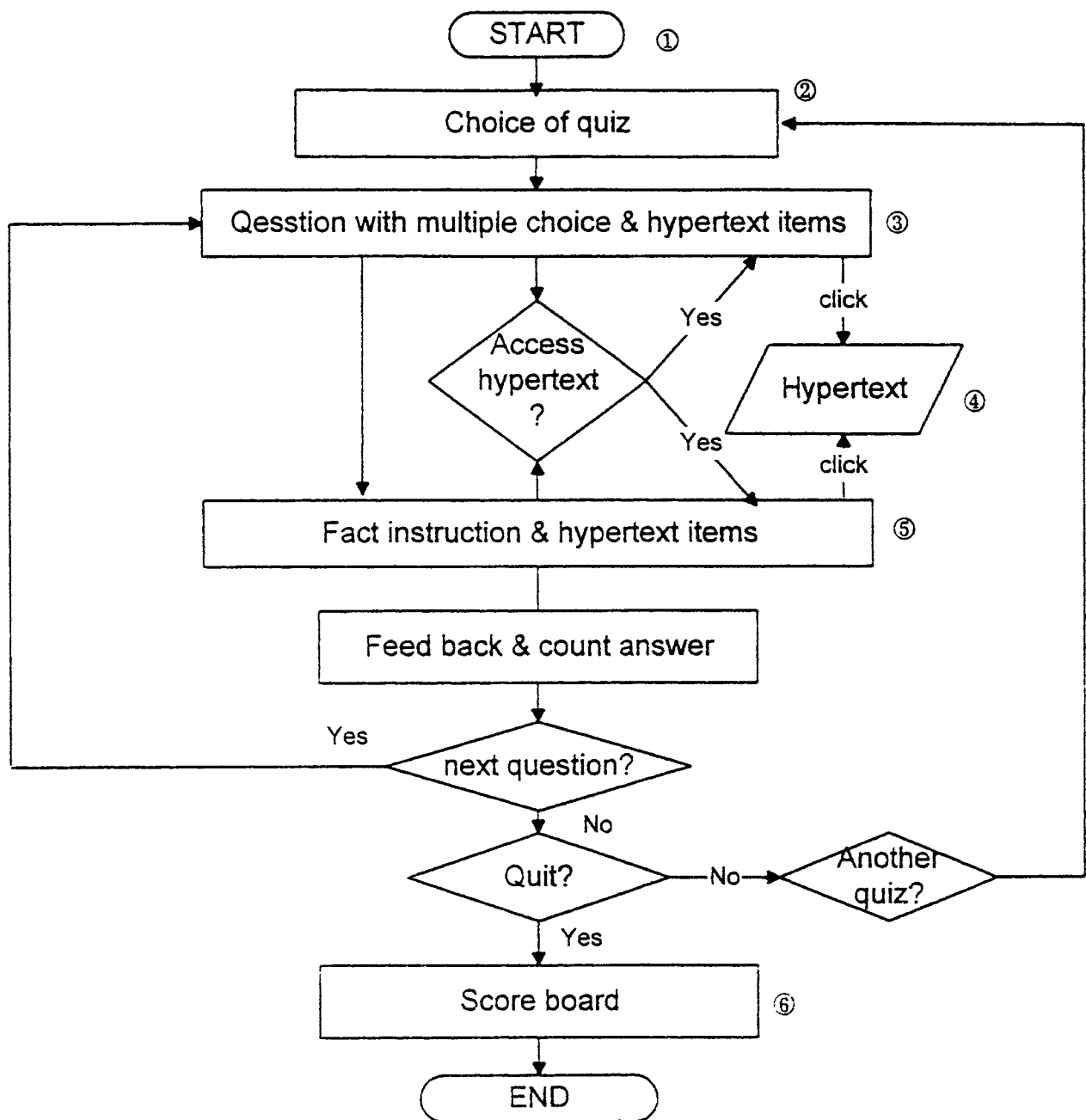


Figure 5. Control Flow Chart for NRKS

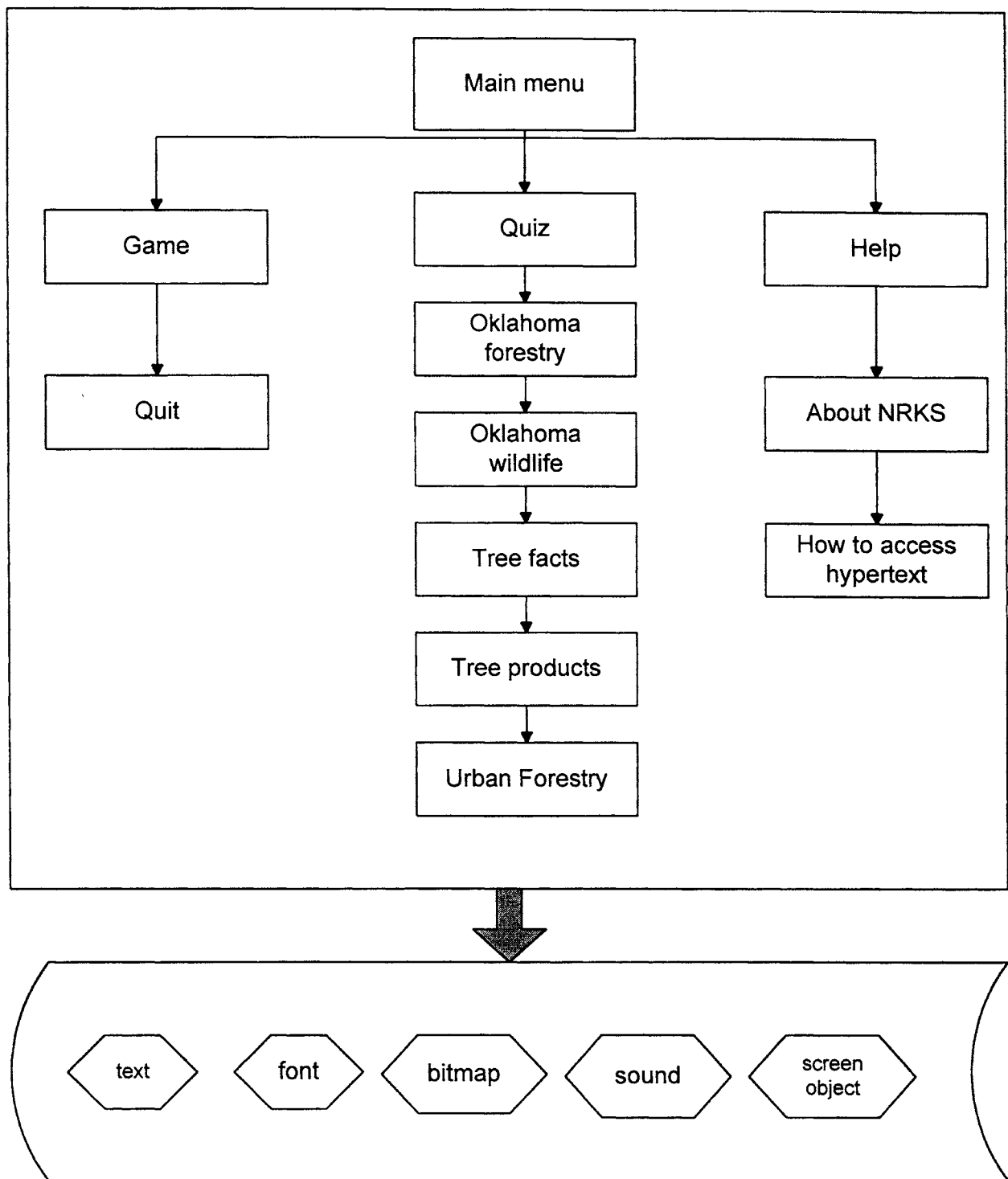


Figure 6. Menu Organization (From System Point of View)

Figure 6 shows the organization of the menu from the system point of view. The gray arrow represents paths to the data objects, namely text, font, graphics, sound, and screen object (button, window, scroll bar, etc.). Interaction between components is by message passing.

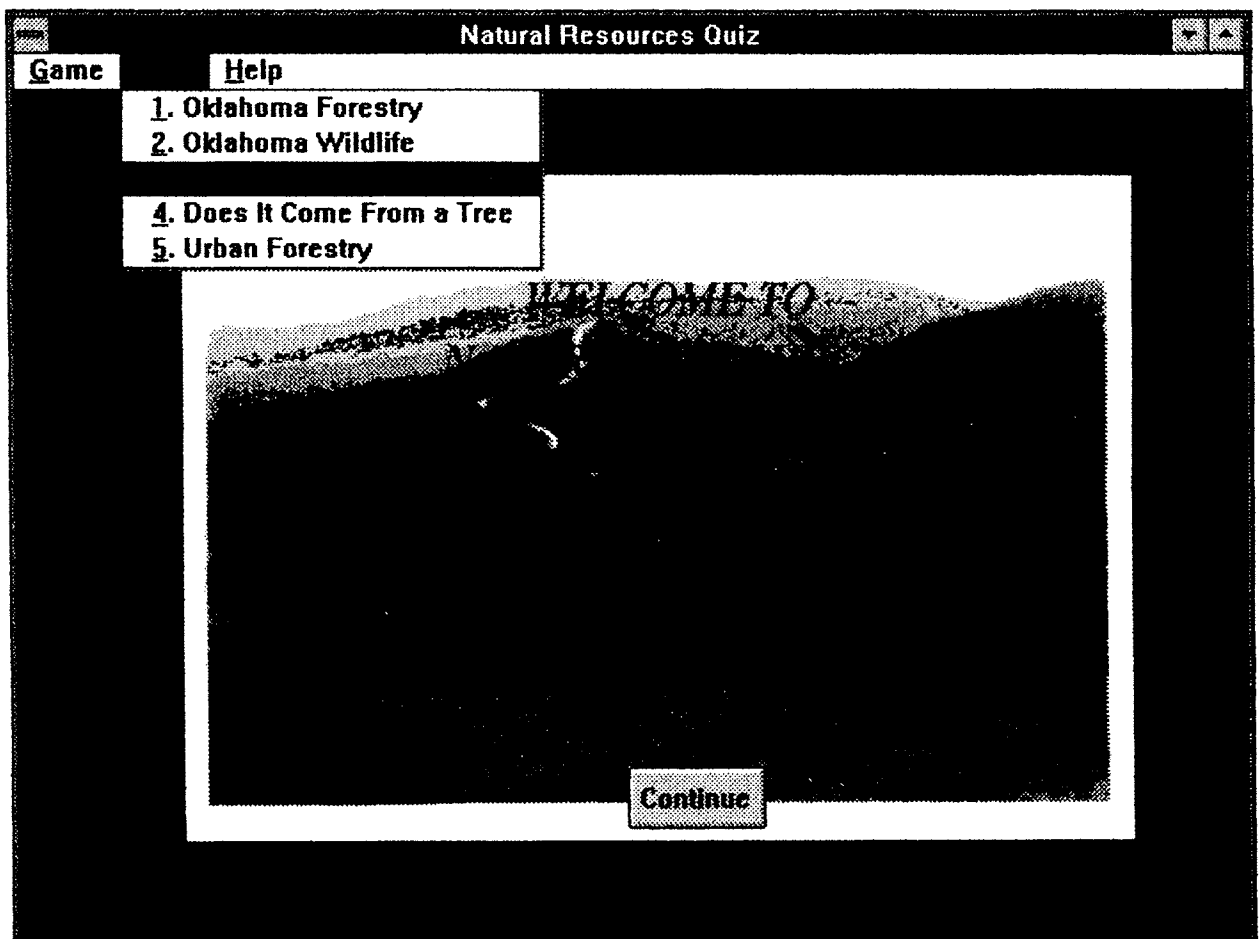


Figure 7. Menu Window (From User Point of View)

User sees the menu screen shown as Figure 7, at the same time, he or she hears a music along with greeting speech. After the user selects one of the quizzes, and clicks the

continue button, a particular path will lead to the quiz selected, and come to part ③ of the control flow chart.

The part ③ consists of two separate windows, which contains question with multiple choices and hypertext items respectively. The two views of window setting for this part are designed as Figure 8 and Figure 9.

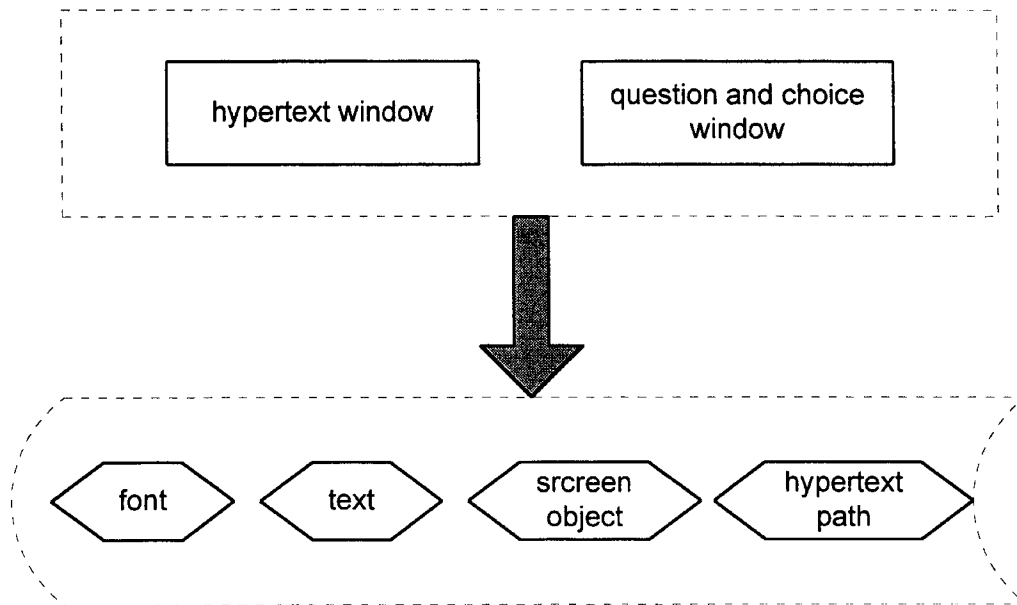


Figure 8. Question Window and Hypertext Window (System Point of View)

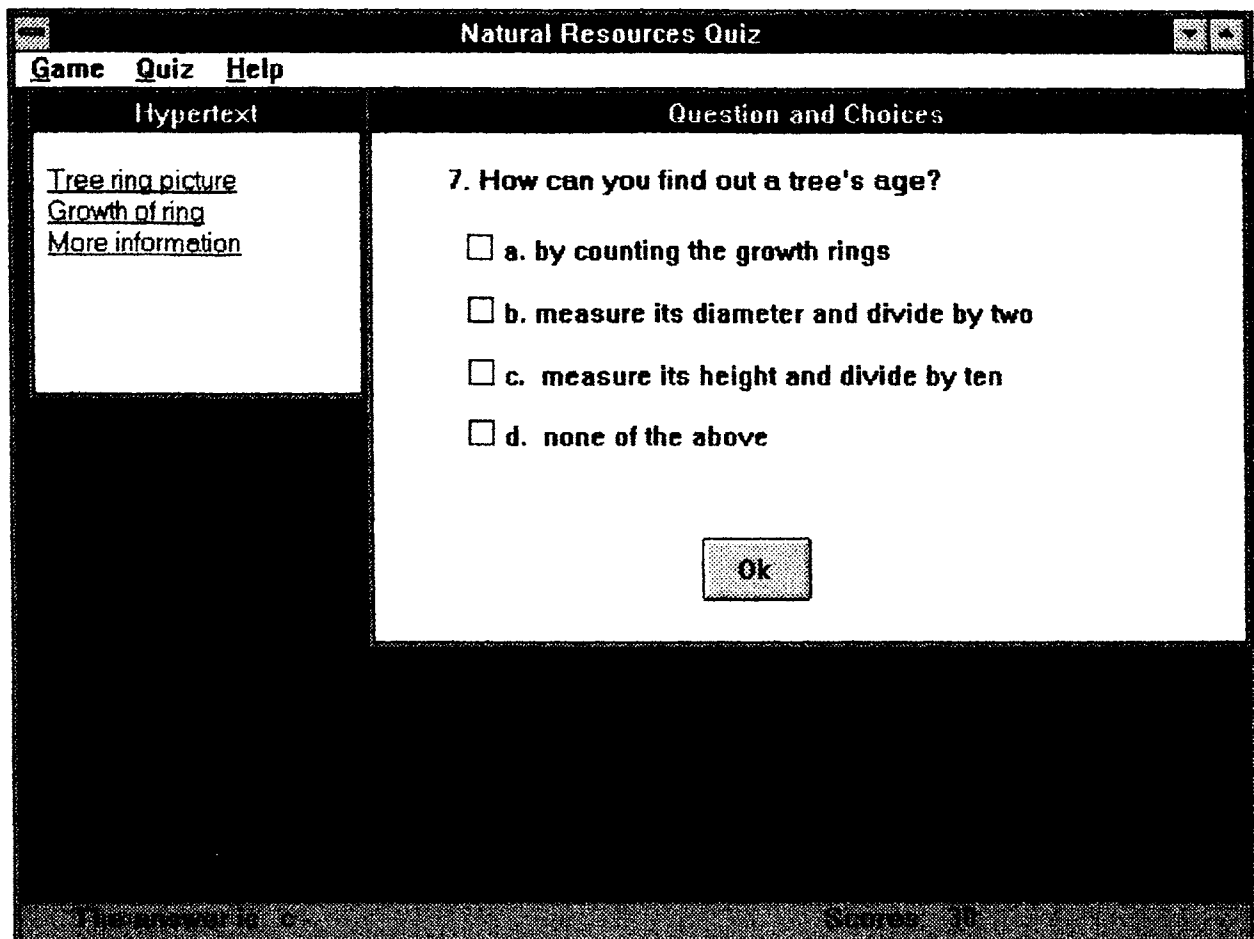


Figure 9. A Sample of Question and Hypertext Window (User Point of View)

The user has certain control over the NRKS CAI. In the screen shown in Figure 9 (this applies to all other questions & hypertext windows), he or she can either click on one of the hypertext or click on one of the answer choices. If the first choice has been made, it will lead to part ④ (Figure 10), otherwise it leads to part ⑤ (Figure 11).

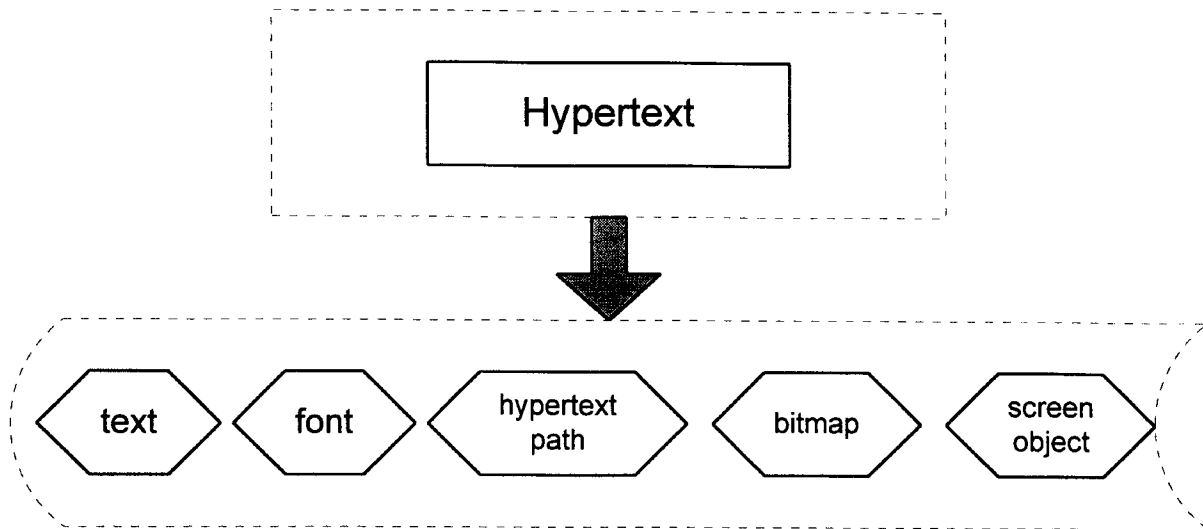


Figure 10. Hypertext (System Point of View)

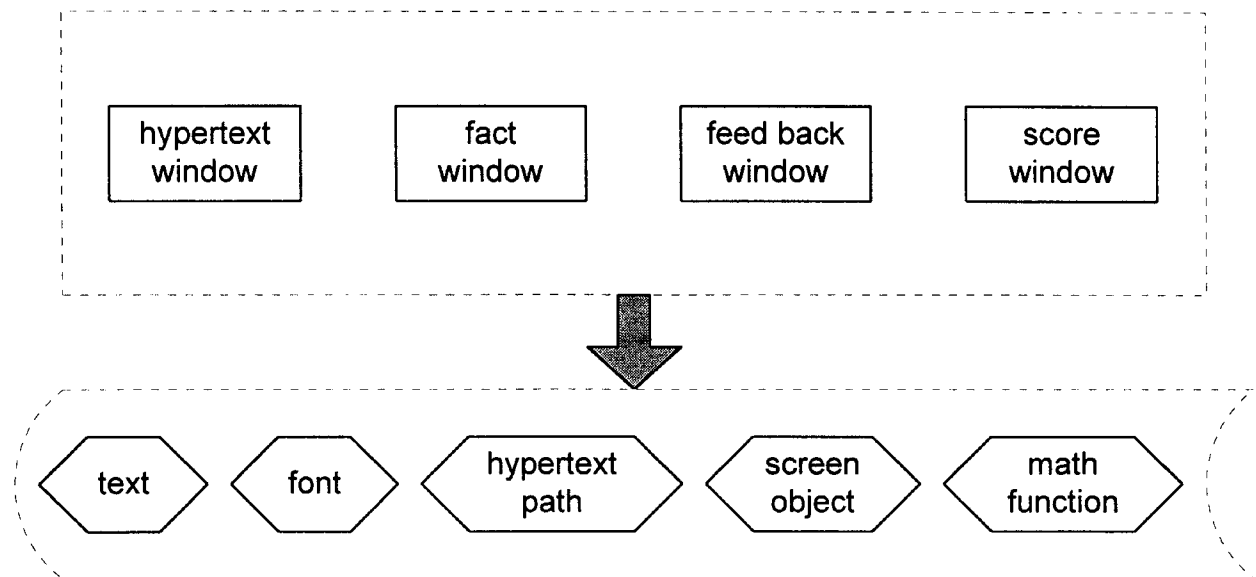


Figure 11. Fact, Hypertext, and Feedback (System Point of View)

For the particular question in Figure 9, the user may click on hypertext: *Tree ring picture*, *Growth of ring*, or *More information*. The user view for these hypertext are shown in Figure 12, Figure 13, Figure 14, and Figure 15.

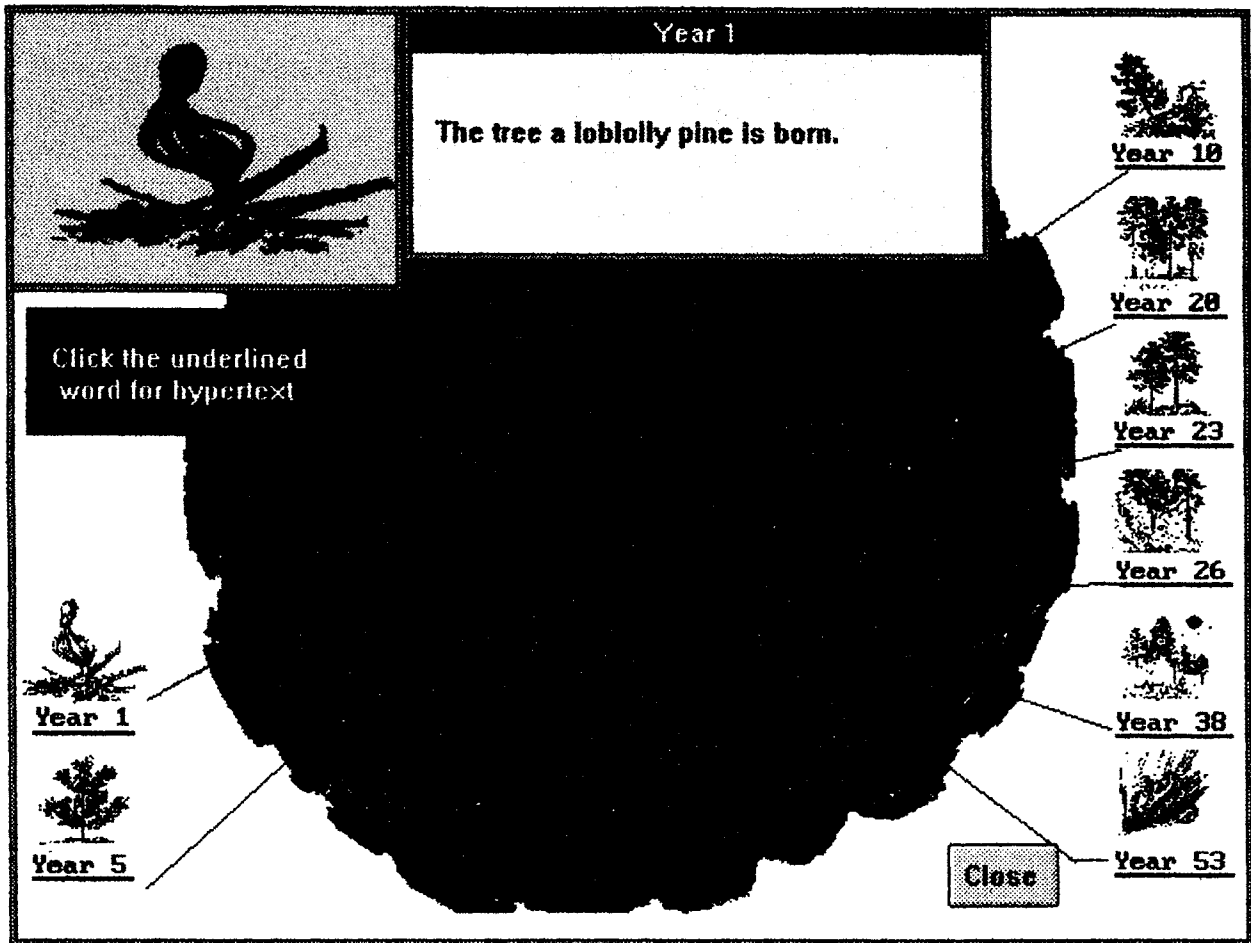


Figure 12. Hypertext for Tree Ring Picture (Hyper-Region on Year 1)

Figure 12 shows a screen reached by the link between hypertext *Tree ring picture* and this graphic screen. This screen contains 8 hyper-regions that show the tree ring growth from year 1 through year 53. If the user clicks on either of these hyper-regions, additional information will become available. The result of clicking on hyper-region "year 53" is to bring out a screen which is shown in Figure 13.

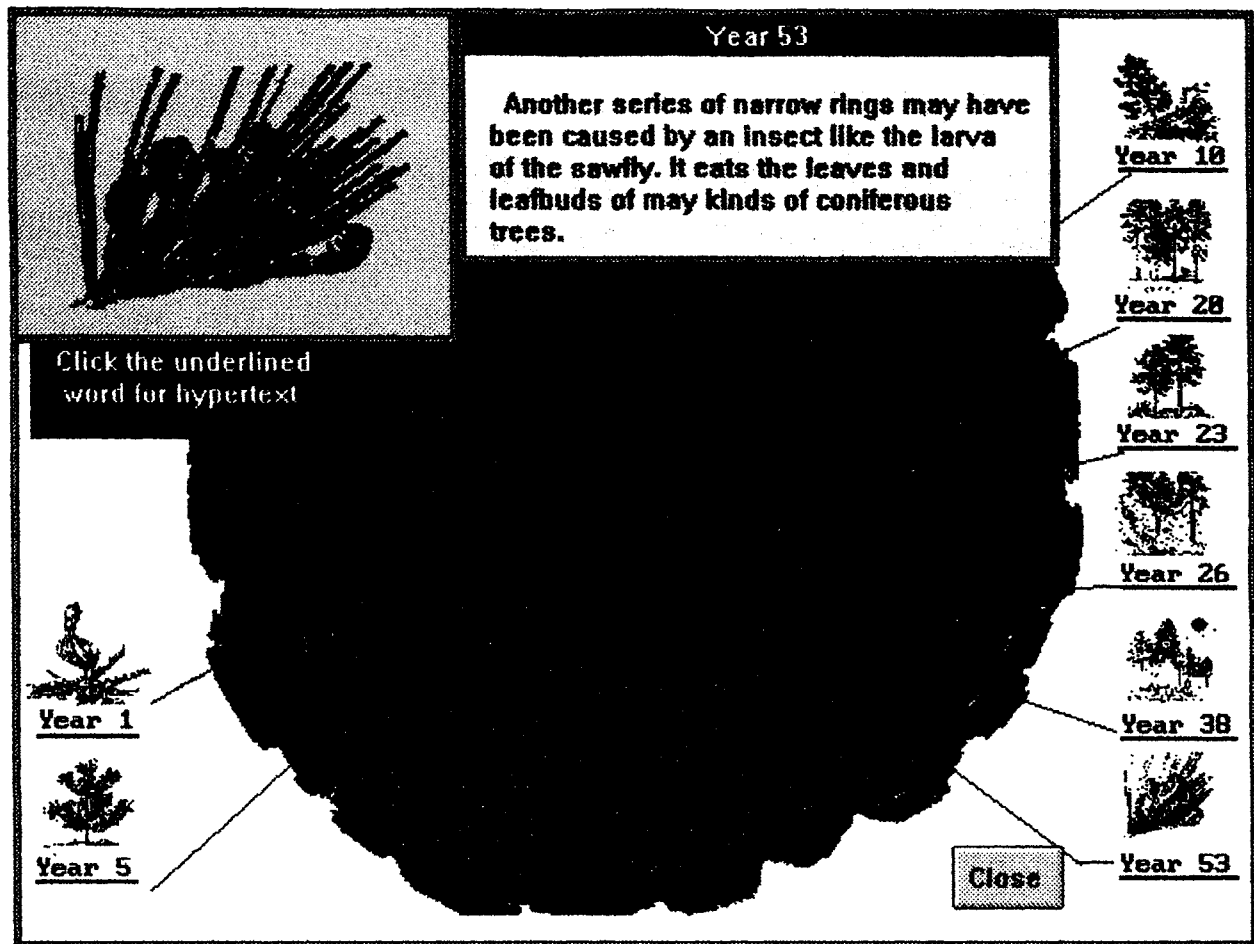


Figure 13. Hypertext for Tree Ring Picture (Hyper-Region on Year 53)

The instruction in blue window tells the user that every underlined word is a hypertext, which contains more information about the ring at that year. The screen shown in Figure 13 tells the user the reason why the tree ring is extremely narrow at year 53. All the hyper-regions together is necessary to help the user get better understanding of the tree ring.

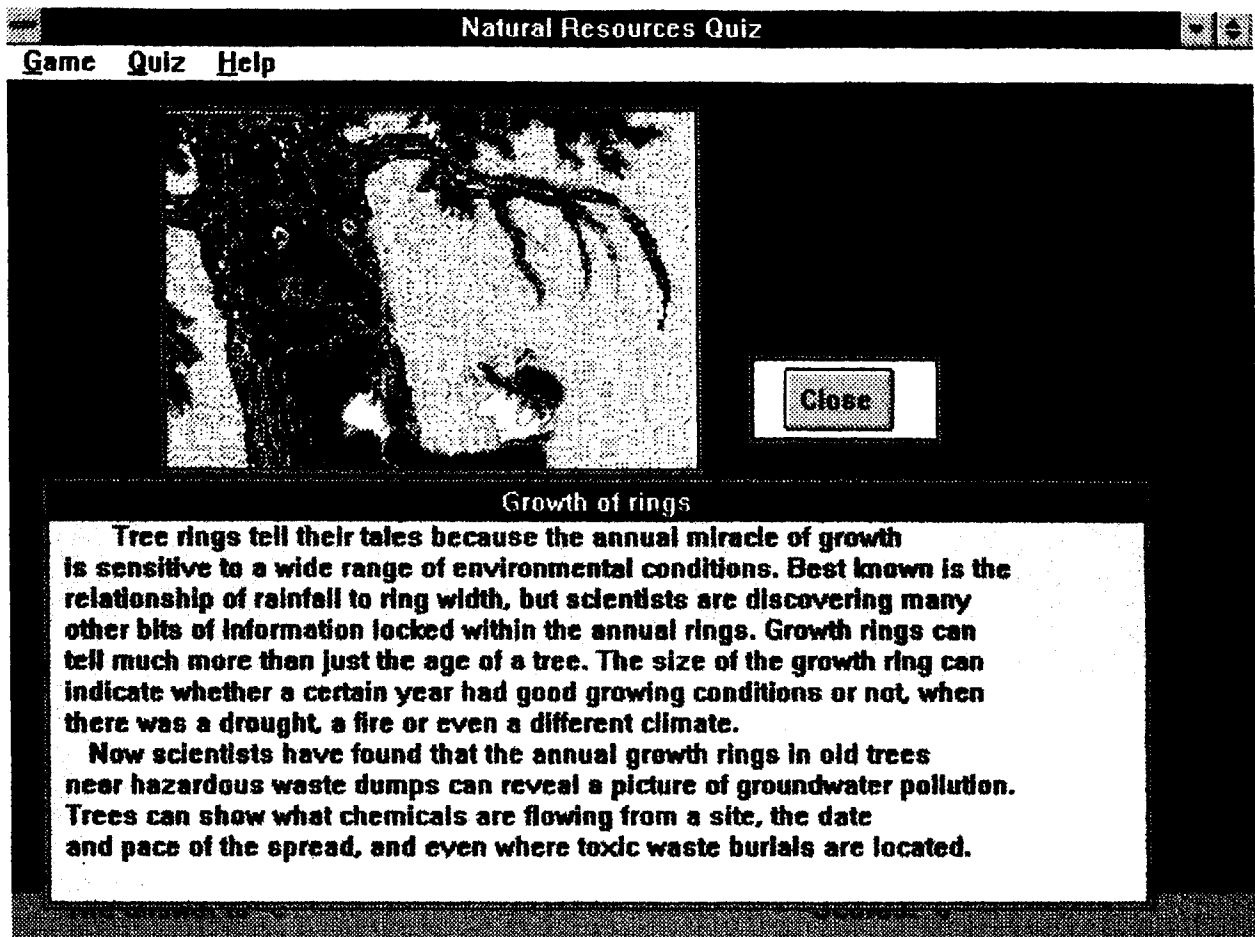


Figure 14. Hypertext for Growth of Ring

The screen shown in Figure 14 is reached by the action of clicking on *Growth of ring* in the hypertext window shown in Figure 9. The user has the control of the display time, after finished reading, he or she can simply click on the close button to go back to the previous screen (Figure 9). Then, the user can access all the hypertext in this question again. It can repeat as many times as the user wants to try.

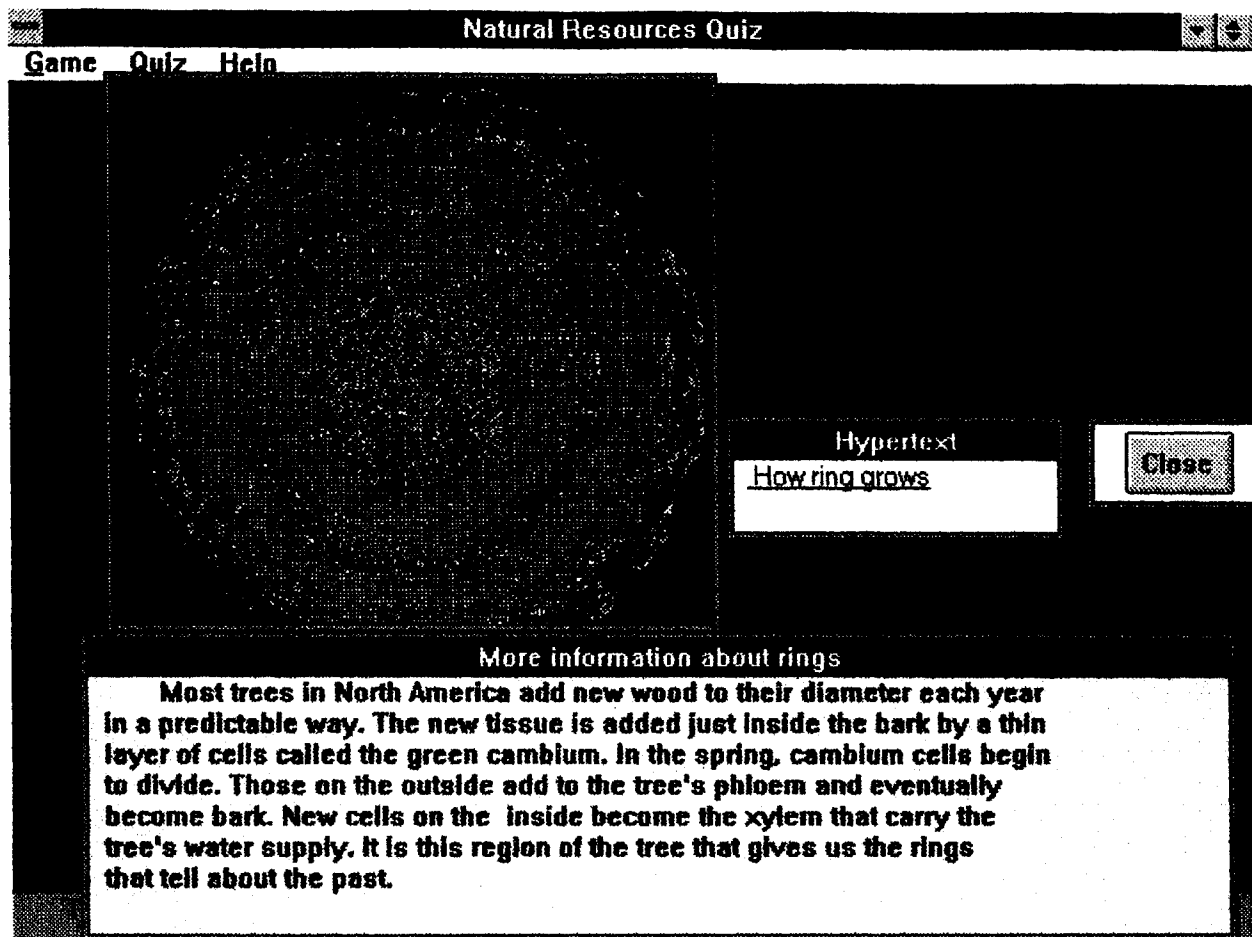


Figure 15. Hypertext for More Information about Rings

The screen shown in Figure 15 is the user view for hypertext *More information*, it contains lower level hypertext (*How ring grows*), which works in a similar way. As mentioned before, in the question and hypertext setting shown in Figure 9 , the user also may click on one of the possible answer choices, in that case control goes to part ⑤, which contains fact, hypertext, and feed back with the score and correct choice shown at bottom of the screen. The user's view of fact, hypertext, and feed back are shown in Figure 16.

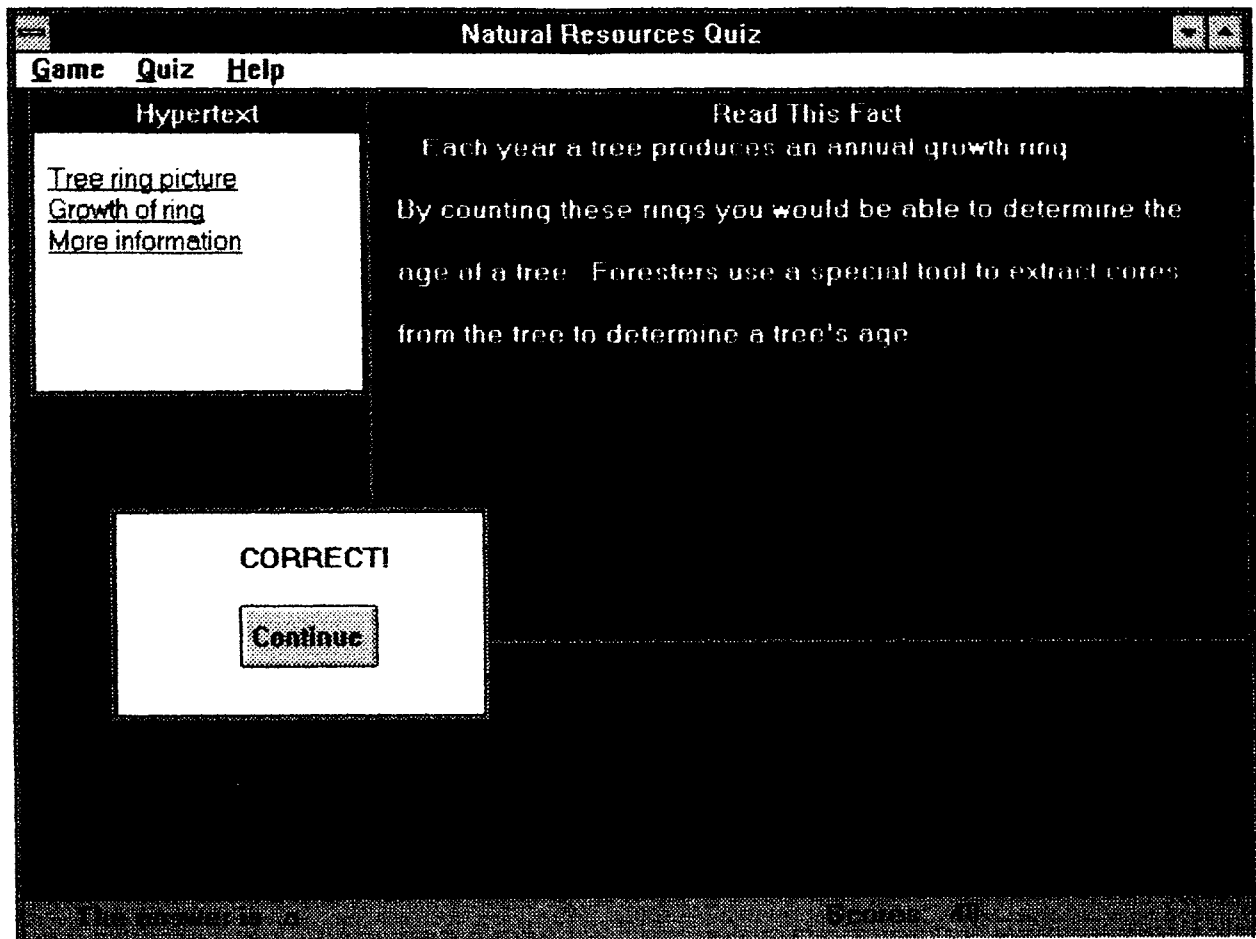


Figure 16. Fact, Hypertext, Feed Back Windows (User Point of View)

At any time, if the user chooses "exit" from the menu, it allows the user to leave NRKS with a score report (part ⑥ in the control flow chart figure). The two views of the score reporting window are given in Figure 17 and Figure 18.

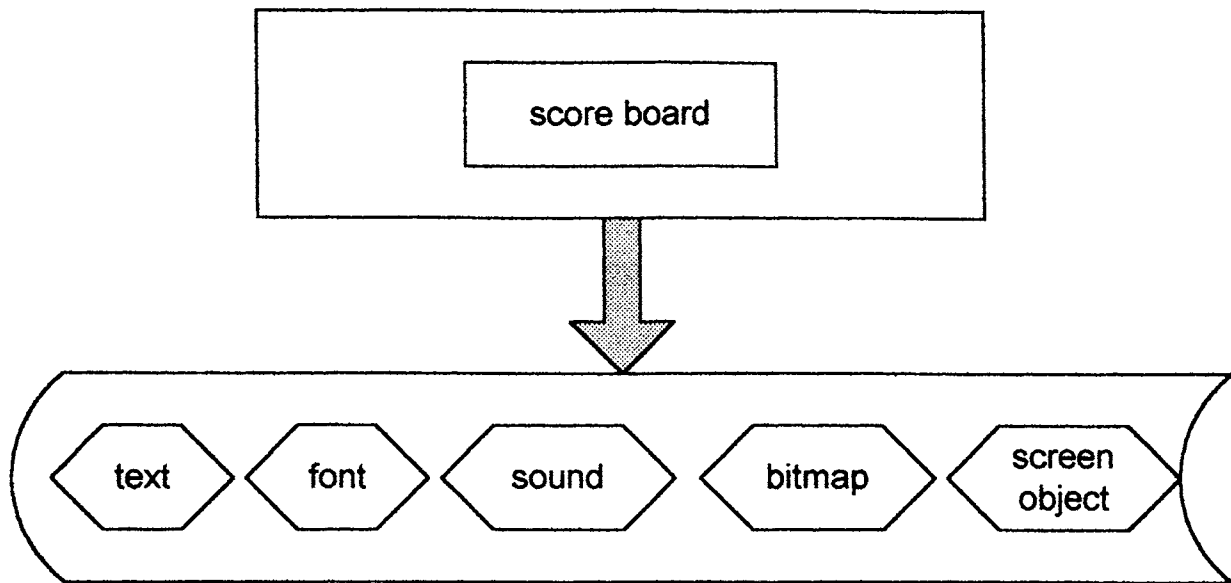


Figure 17. Score Reporting (System Point of View)

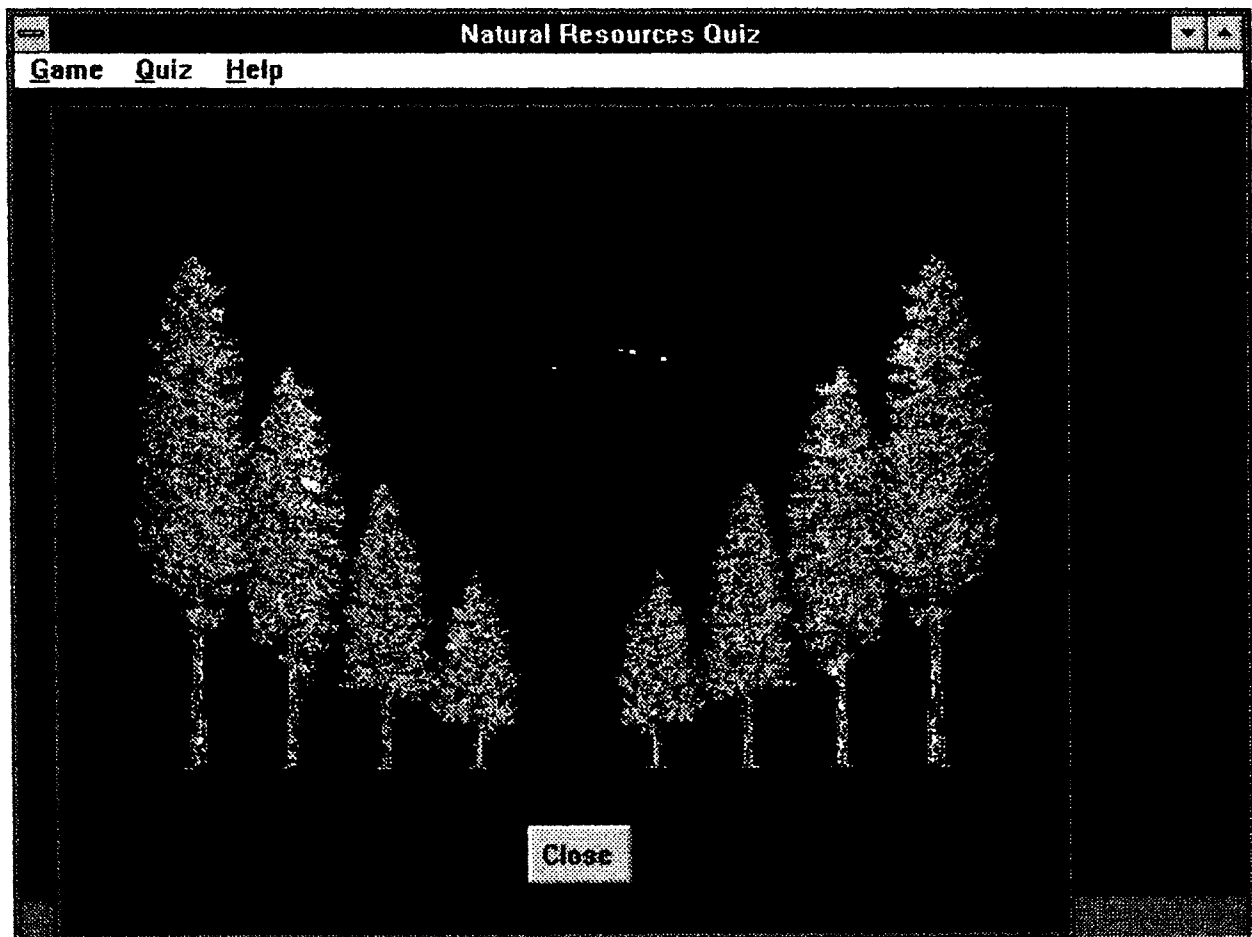


Figure 18. Score Reporting Window (User Point of View)

2.2. Software Design via Knowledge Pro

2.2.1. Knowledge Pro Environment

Knowledge Pro is a powerful programming environment, it combines the following technologies: Object oriented programming, Hypertext, Expert systems, List based programming , Visual programming. Knowledge Pro is a development environment which allows the user to build complex windows applications rapidly. The environment is made up of:

- a rich object-oriented programming, and list processing language
- design tools for point and click design
- a librarian and library of topics to extend the language
- couple of powerful debugging tools
- a multi-document editor
- a flexible, customizable help system

The heart of the environment is the Knowledge Pro language, which combines OOP and list processing programming into one environment.

2.2.2. Software Architecture of NRKS

Topic is the most important concept in Knowledge Pro, and it is the basic unit of software architecture of NRKS. All applications are collections of topics. A topic is a block of code which can serve as object in OOP. We complete every task by sending message with necessary information to topics. A topic can be called by name, or by do command, or by hypertext and hyper-region, or by screen events, or by inheritance. The architecture and organization of topics will impact an application.

This section presents the topic structure of NRKS. The top level of the NRKS topic structure is shown in Figure 19.

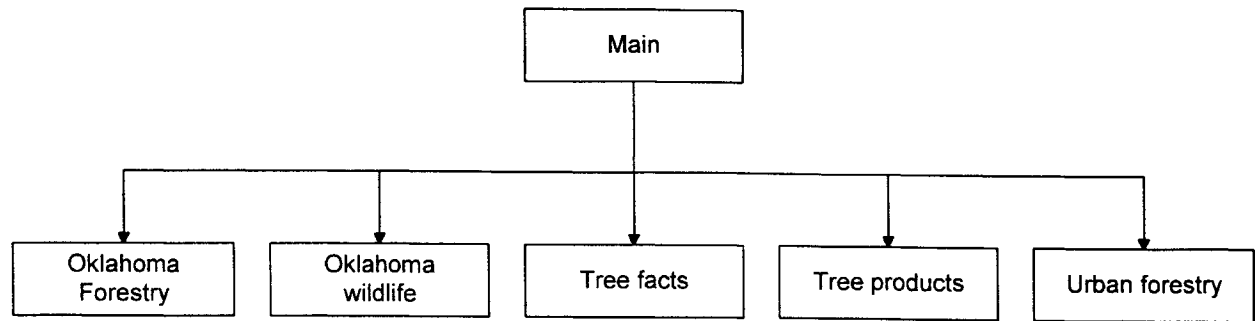


Figure 19. Top Level Structure of NRKS

The five children of the above tree structure represents the five components of NRKS. They are implemented as Knowledge Pro topics, and correspond to the user interface menu items.

The second level of the structure applies uniformly to these five topics. Here we give an example for "Treefacts" topic in Figure 20.

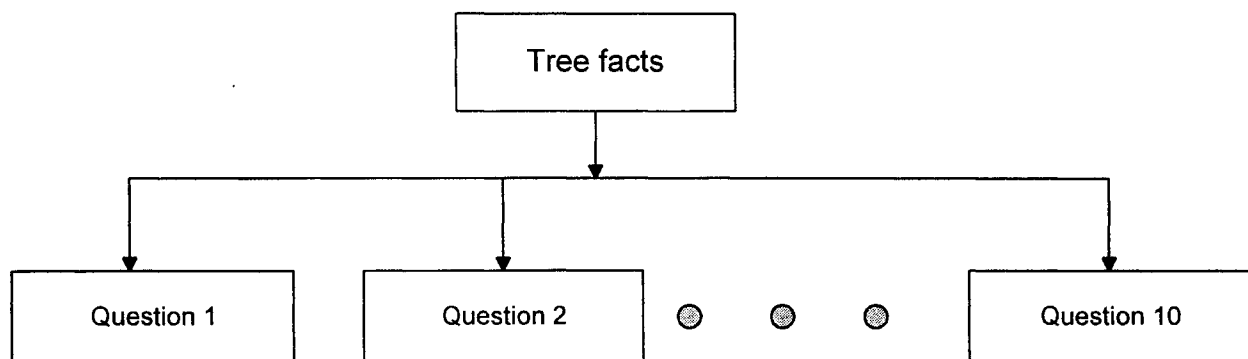


Figure 20. The Structure of Treefacts

The Treefacts topic consists of ten questions. Each question is represented by a topic. Each of these topics is further constructed from children topics. The structure of a question topic is shown in Figure 21.

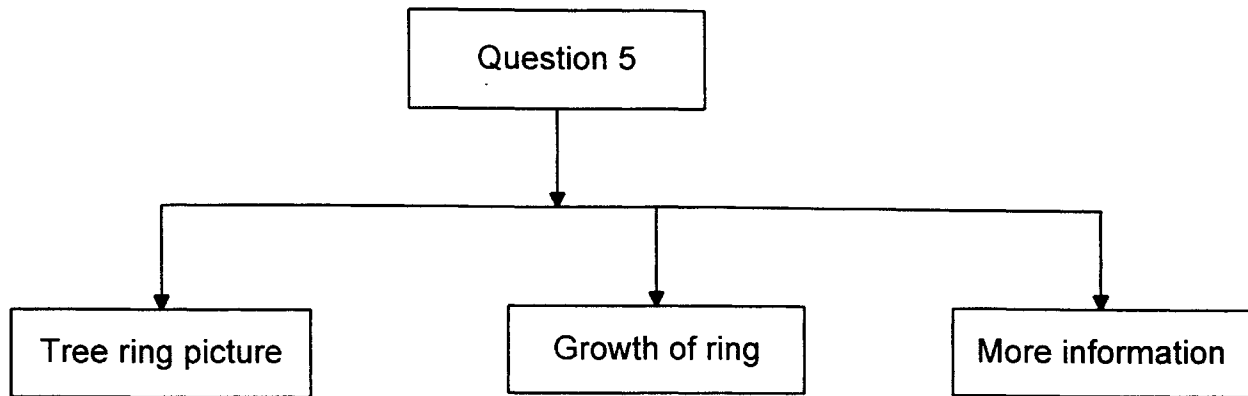


Figure 21. The Structure of a Question in Treefacts

Usually every question topic contains some hypertext topics, the topic *Question 5* in Figure 21 contains three hypertext topics: *Tree ring picture*, *Growth of ring*, and *More information*. Sometimes these hypertext topics even can be decomposed into their children topics. This situation is illustrated in Figure 22.

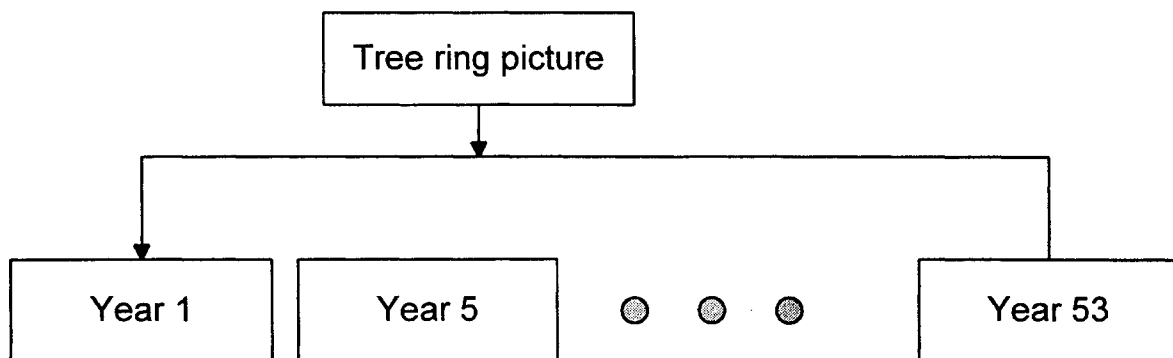


Figure 22. The Structure of Children Topics of Hypertext

In common situations, a task is not completed by a single topic. Usually the topic calls other topics to do part of the job or to get some information such as text data, screen object, the corresponding feed back, etc.

2.3. Screen Design

2.3.1. Importance of Development Environment

During the development of NRKS, several software are used to deal with the graphics. We know that every software package needs a driver, or more than one driver. If two drivers can not reside in memory at the same time, a problem is created. Good understanding of video card standards are required to ensure smooth interaction of software developed using different packages. Figure 23 and Figure 24 are provided to illustrate the importance of understanding rule and operation of a video card. Figure 23 and Figure 24 are the two screen layouts when the same program runs under two different system settings.

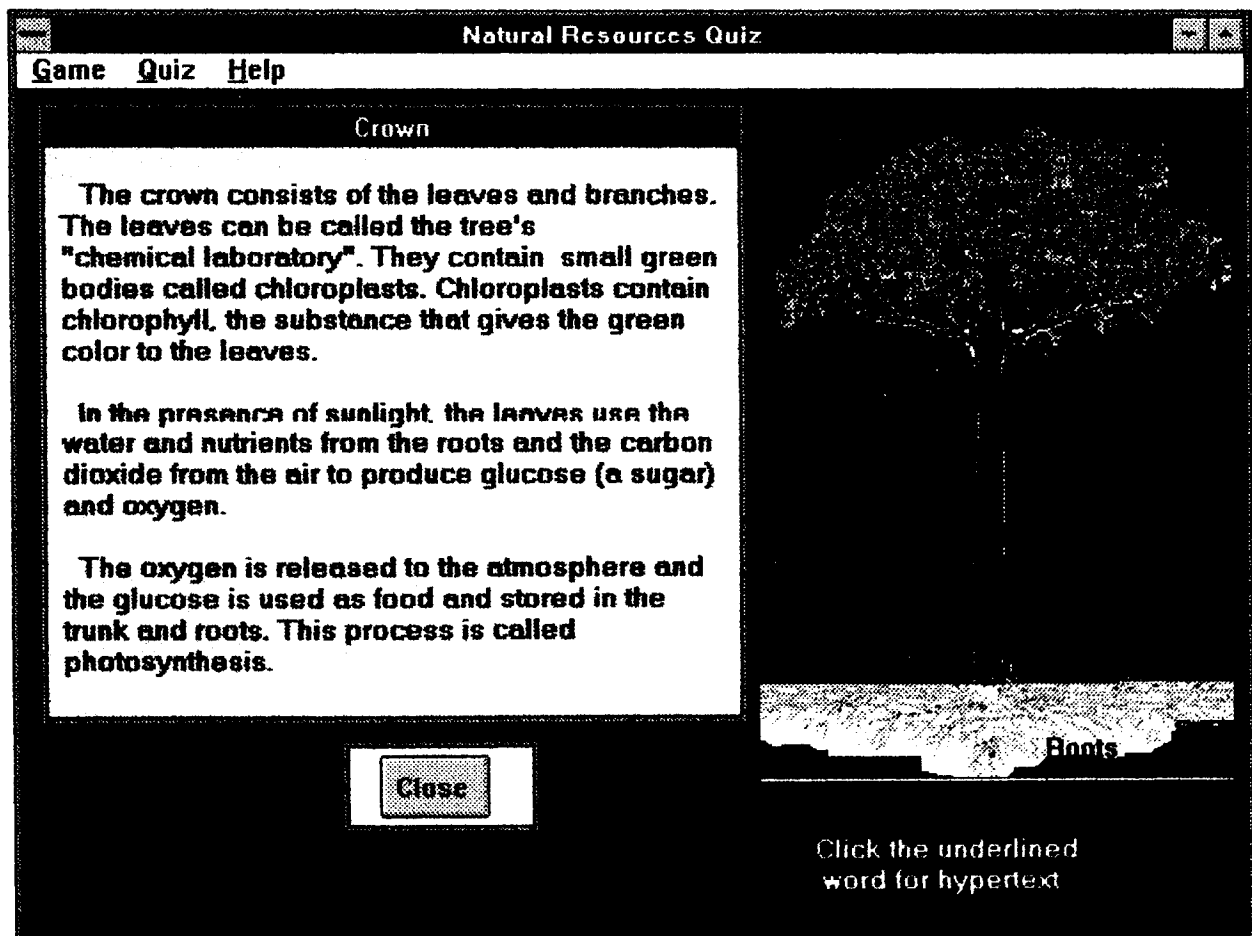


Figure 23. Screen Layout in Resolution 640×480

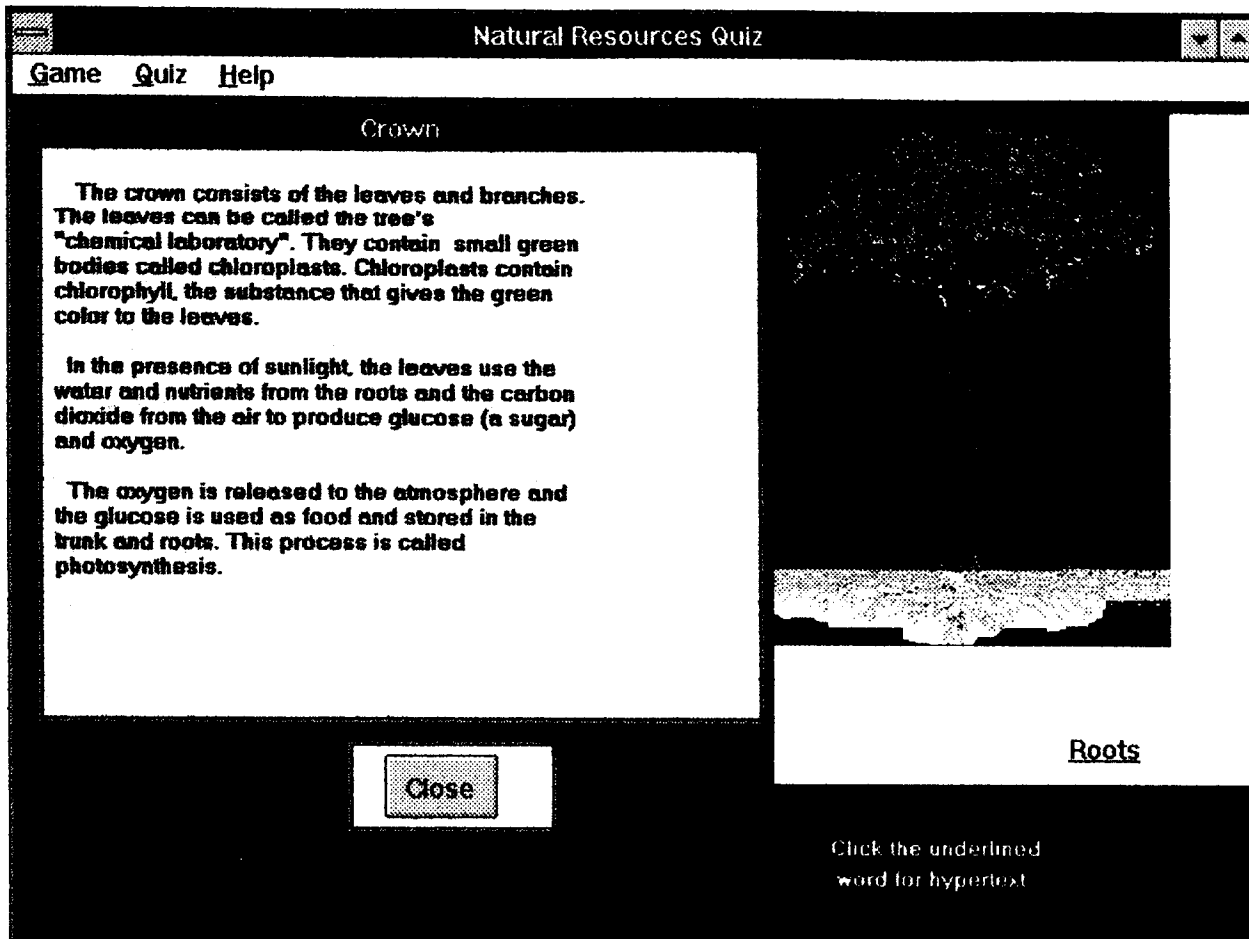


Figure 24. Screen Layout in Resolution 800×600

The reason why the text and the tree picture in Figure 24 look awkward is because of the different ways they are mapped to the window. The text is written to a particular position in the window which will display tree picture. The text position fits the picture well at resolution 600×480 as shown in Figure 23. But if the resolution changes, then the picture size changes, which causes the text position no longer to fit in the picture smoothly (as show in Figure 24). For this particular problem, we can change design mechanism to make text combined with graphics. The picture and the text working together as one graphics file will prevent this kind of problem. We also can use `system_info` command to get the necessary system setting information in the application,

to help in making appropriate displays. The designer must take into account the system setting when creating programs.

2.3.2. Graphics Display Methods

The DOS version and the Window version of NRKS use different kinds of graphic files, the first version uses PCX files, and the other one uses BMP files. The method used to bring the graphic files to a program is also different for DOS and Window in NRKS. We will discuss the two different methods in this section.

In the DOS version of Knowledge Pro, in order to bring a graphic object into a program, a graphics toolkit which comes from the same company has to be applied. This toolkit is a software designed to make a topic which contains the desired graphics and all the hyper-region (the region can lead to more information, either in text or in graphics form). In the application program, the topic is called to show the desired graphics with the hyper-region in the screen. An example of graphics topic taken from graphics toolkit is given in table IV. An application program can call the topic.

TABLE IV
CODE SEGMENT FOR PICTURE CROWN TOPIC

```

topic crown.
select is element (picture ('c:\treefact\pcx\CROWN',1,8,N, ( path of the graphics file )
    [['trunk',G,114,266],                ( hyper-regions )
    ['bark',G,164,328],
    ['roots',G,125,429]]),2).
if ?select <> "
    then do (?select).
end.
```

The topic shown in table IV specifies path name of the desired graphic file which is *c:\treefact\pcx\crown*, and it defines three hyper-regions which are related to topics: *trunk*, *bark*, and *roots*. When either one of the three hyper-regions is clicked, the corresponding topic will be called and executed.

All the coding work (including the graphics and icons) in NRKS for Window are completed in Knowledge Pro for Window environment. BMP files and icons work in similar manner. The commands that work with these two kinds of graphics are *load_bitmap* and *load_icon*. Some other commands are also usable when we deal with BMP files. For example, the *bitmap_info* command will return a list which contains the width and the length of the bitmap, then we will decide the size of the window in which we are going to put the picture. We also can use some commands to resize a bitmap, save a bitmap, print a bitmap, copy a bitmap to the clipboard, and so on. In both cases which involve icon or bitmap, after the topic which contains an icon or bitmap has been executed, we need to delete the icon or bitmap before exiting the topic. This step will free the memory no longer needed by the program and make programs run faster.

2.4. Audio Application Design

NRKS with sound application must have hardware support (e.g. a sound card). NRKS is developed in a computer with Sound Blaster. All the sound files (WAV files) are either recorded or mixed by using Sound Blaster software package.

Similar to dealing with graphic files, NRKS DOS and NRKS Window have to apply different methods to implement the sound files. In DOS, we use DOS command to run the external program which play the sound within NRKS application. The dos command as the format: *dos ('WPLAY c:\kpwing\samples\kgintro.wav')*.

For NRKS Window, in order to play sound files, we have to access Dynamic Link Libraries (DLL), which allows calling external functions from Knowledge Pro. A section of code is given in table V for illustration.

TABLE V
CODE SEGMENT FOR SOUND APPLICATION TOPIC

```

topic wel_sound.
  command_list= ['open c:\kpwing\samples\kgintro.wav alias sound type waveaudio',
  'play sound wait',
  'close sound' ].
  lib is load_library ('mmsystem.dll').
  while ?command_list <> [] then
    call (?lib,mciexecute,[pointer (address (first(?command_list) ))],int)and
    command_list=rest (?command_list).
  free_library (?lib).
end. (*wel_sound*)

```

If the topic *wel_sound* listed in table V is called, the sound file *kingtro.wav* under directory *c:\kpwing\samples* will be played immediately.

2.5. Dynamic Visual Display (DVD) Design

DVDs may refer to any kind of moving graphics. DVDs on a computer is very attractive to CAI lesson designers. In this thesis, two kinds of DVDs are incorporated in NRKS. Again, the mechanism to implement DVDs is different in DOS and in Window environments. First, we will discuss the mechanism working in DOS environment.

How to make graphics moving? Several software packages are available to do this. One of them is PC Animate Plus. A sequence of picture frames are created by either drawing color pictures, or by loading available pictures. The moving direction and speed of these frames can be controlled by a user in PC Animation Plus. Also we can combine sound and the moving graphics together into one animation file. Animation file is created independently and included in NRKS. To run the animation produced by PC Animate Plus, *animate.exe* file (this is a free distribution file) has to be copied to the application. At the point we want to run an animation in NRKS, we have to halt the execution of the current knowledge base and run the external animation program. We have two choices to do this, either by dos command or by run command. The following is an example of dos command: *dos ('animate c:\kpwing\win_quiz\tutorial.ani')*.

In the MS Window environment, we even can play Video with sound. Some of hardware and software are available in the market to capture Video from video devices such as VCR and Video camera, then digitize them. VideoSpigot for Windows and Video Blaster are two products from Creative Labs. These packages can convert Video tape to a file such as an AVI file which is computer readable. How can NRKS read the AVI file? Similar to a sound file, we must export Video application from DLL. An example video topic is given in table VI. When we need to execute this topic, we send a message to this topic. Then the video file *new.avi* contained in *c:\kpwing\samples* directory will be displayed.

TABLE VI
CODE SEGMENT FOR VIDEO APPLICATION TOPIC

topic Video.

VideoWin is window (,8,15,35,20,,[popup,thinframe],?main_window).

set_text (?VideoWin,').

TABLE VI (Continue)

```

lib is load_library ('mmsystem.dll').
command_list= ['open c:\kpwing\samples\new.avi alias video type avivideo',
'play video wait',
'close video wait' ].
lib is load_library ('mmsystem.dll').
while ?command_list <> [] then
call (?lib,mcieexecute,[pointer (address (first(?command_list) ))],int)and
    command_list=rest (?command_list).
free_library (?lib).
end.(*Video*)

```

2.6. User Interface Design

User interface is very important to the success of a CAI lesson. It decides the way how the user is going to communicate with the lesson. A good user interface makes it easy to gather information from the user, and presents tailored material to the user's special needs. User interface usually refers to screen objects and their grouping. When we build the user interface, we should pick up the screen objects that will best suit for the information being gathered. When some information are needed from the user, usually we may provide either one of the following three types of input.

1. enter a value
2. choose one or more items from a selection list
3. select an available action to perform

For every one of the three input types, there is a suitable group of screen objects associated with it. In this section, some consideration of the relation between input types and the screen objects in NRKS will be provided. We will discuss this in the order from input type 1 to input type 3. In NRKS, the information such as the user's name and age is gathered by entering a value. This is accomplished by providing an edit line in the screen and letting the user type the information. Other screen objects such as edit box, edit window, and edit file will be good choices when the length of the input information suits to them. The second input type is heavily used in NRKS. Multiple choices are provided to every question in the quizzes, the user need to choose one item from the multiple choices. List boxes, combo boxes, check boxes and radio buttons are the available screen objects to implement such a task. Which one will be the best one depends on how many choices are allowed by the task, and on the number of the total available choices. The possible answers for questions in NRKS are presented in check boxes. The last input type is selecting an action to perform. By this way, the user can control what happens next. Some examples of this kind of input type can be found in NKRS. The menu in Figure 7, the continue button in Figure 7, and close button in Figure 11 are three examples.

CHAPTER III

SOFTWARE EVALUATION

3.1. Necessity of Evaluation

In Chapter I, we discussed evaluation and modification between every phase. Here, we focus on the evaluation which would be necessary after development phase. At this point, we need to consider different factors related to attitudes, curriculum priorities, learning environment, etc. Evaluation may be a time consuming process, but it is a very necessary process to improve the effectiveness of a CAI software. Evaluation [Hannafin & Peck, 88] is an ongoing process used to determine whether the goal has been achieved, identify which part of the lesson needs to be modified, and appropriate modifications. There techniques to evaluate CAI lesson, belonging to two groups: formal techniques and informal techniques. Both techniques can give the evaluator unique information, based on which they can determine what kind of revision is necessary. But formal techniques require sophisticated techniques and procedures which are usually not familiar to programmers and lesson designers. This thesis will apply informal techniques. If we consider the levels of evaluation, there exist two levels: formative phase and summative phase. In the formative phase, evaluators identify the lesson features, and point out the necessary modification, after the modification has been complete, reevaluate the lesson again. In the summative phase, the evaluators usually reach the final decisions such as the value of purchasing system on the CAI lesson. In this thesis we are only concerned with the formatives phase.

Dick and Carey [Dick & Carey, 85] provided a three stage formative evaluation scheme based on the stages of development of the lesson. The three stages are one-to-one evaluation, small-group evaluation, and field test evaluation.

One-to-one evaluations usually apply during initial lesson design and development. At this stage, not many materials have been put into the lesson, but the general setting of the lesson is available for test. The test population may include the designer and a student, or the designer and a teacher, or the instructional designer and the programmer. They go through the test together providing valuable information about the lesson design before significant resource are put into the lesson.

Small-group evaluation usually applies when the lesson is almost finished. A small number of students need to participate in the lesson test. The purpose of this stage is to identify where the lesson needs improvement, and to determine the effectiveness and acceptability of the CAI lesson. The NRKS had been taken to a primary school in Stillwater, Oklahoma to make a small-group evaluation, about 10 students went through the lesson. We gathered data on the students' performance and the comments from the students and the teacher. The information we gathered include the answers to the following questions such as: Can you easily follow the instruction to run the software; Which part of the lesson is difficult to you; Do you like the lesson; etc. All these information from the students and the teachers is valuable for us to improve the lesson.

Field test evaluation is generally conducted when the lesson is almost in final form. At this stage, only minimal additional modification will be made if it is identified as necessary.

3.2. Components of CAI Evaluation

3.2.1. Evaluating Instructional Design

The principal components we considered are listed below:

- The outcomes of the lesson are as specified in the objective. We can tell this from the feedback of the users. NRKS provides automatic score recording, if the average score is good, it means the users have achieved the most goals of this lesson.
- The content of the lesson is accurate and complete. The material in the five different quizzes in NRKS are designed by different professors who are experts in these particular fields.
- The level of difficulty (include the reading level) is appropriate to the target population. Dr. Champe Green and Bille Chambers are members of our design group, they also are child education specialists. A software named Right Writer is used to check the difficulty and reading level of NRKS.
- The instructional sequence is reasonable and helpful to finish all the tasks, and is easy to follow. In NRKS DOS, every screen has instructions to inform the user of the possible screens that follow. For the Window version of NRKS, the user has more control over the program. Besides some instructions similar to those in NRKS DOS, the user may access the help menu at any point when running the program.
- Important concepts, and information are amplified effectively. The large number of hypertext used in NRKS are designed to meet this requirement.
- The activities, and procedures of the lesson are likely to motivate students. The audio and animation in NRKS are mainly for motivating the user.
- Feedback on student response is quick, and provide adequate reason. After the user chooses an answer, the feedback will be given to the user immediately with a fact window in NRKS.
- The format (use game, simulation, or other format) of the lesson is appropriate to the material and the educational need. NRKS chooses game format based on the material and the characteristic of the target population.

- The lesson provides students certain level of control options. In NRKS, students can use menu to access help information or to review the pervious lesson, or to quit the system at any point.

3.2.2. Evaluating Program Design

A well designed CAI program should be easy to use, be free from errors, and take full advantage of the unique capability of a computer: interaction, self-pacing, branching, graphics, and sound. What we considered in this evaluation is listed here:

- Instructions for using the software are clear, and consistent, and easy to follow. The students can easily and independently operate the program.
- The program is reliable and works smoothly in normal use.
- Suitable user interface was chosen according to the lesson. For a primary student it is easier to pick a choice in menu than to remember and type a command to control the flow of the program. So, NRKS uses menu-driven form instead of command-driven form.
- Good screen display design. The program should use screen space effectively, and avoid crowding of text in the same screen. When lots of text need to be displayed, we should use either paging or scrolling to make it look neat and easy to read.
- There are three kinds of branching: linear branching, multitrack branching, and regenerative branching. A good CAI program can provide different levels of instruction based on students' responses. NRKS provides multitrack branching, and the student has the control of the direction of branching.
- Sloane said that [Sloane, 89] "Self-pacing lets students proceed at a pace appropriate for their individual learning levels". NRKS is a knowledge based CAI, so we allow the student to control the time for responding by clicking a screen object such as a button.
- Graphics, sound, and animation of the lesson should facilitate the learning process, and can hold the student's attention, make him or her motivated to learn the lesson.

3.2.3. Evaluating Management Design

Management features provide the information about the performance of the students who are taking the CAI lesson, such as record keeping, grading and report generating, etc. NRKS can record every student's performance automatically, and put the information (student name, date, every single answer the student made) to a data file. Then we use a program (written for NRKS) to calculate all the data to give out individual report and class status report. These reports will be valuable for analyzing weak areas of students and class learning, so the designer can rethink about the instructional structure of these areas. The way NRKS deals with the data is called computer-generated management data and report. This method has several advantages, it works quickly, accurately, and it saves labor. The following are some example reports for NRKS.

TABLE VII
SOME SEGMENT FROM RAW DATA

DATE:	4-20-94
Carol:	c b b c b c a b b b 8
Kenneth:	c b d c b b a c c b 8
Jeffrey:	c c d a b c a c c b 7
Robert:	c b d c c a b b c a 7
Charles:	c a a d c d c d a c 1
Andrea:	c b d c a c d b c b 8
Richard:	b d d c c c a b c b 7
Linda:	a b d c b c a b c b 9
Shirly:	d b c c d c a b c b 7

TABLE VII (Continue)

Susan:	c b d d a c a b c b 8
Michael:	c b d c c a c b d d 6

The above table contains some raw data about a student's performance (due to space constraint, we just showed a segment of data for treefact quiz part). The raw data is kept in a data file automatically when a student runs the software. A C program is written to analyze the data, in order to get individual report (Table VIII is an example) and class report (Table IX is an example).

TABLE VIII
AN EXAMPLE OF INDIVIDUAL REPORT

NAME: Carol	DATE: 4-20-94	QUIZ: Treefacts
YOUR CHOICE:	c b b c b c a b b b	
CORRECT CHOICE:	c b d c b c a b c b	
YOUR SCORE:	8 out of 10	
AVERAGE SCORE:	7.4 out of 10	

TABLE IX
AN EXAMPLE OF CLASS REPORT

Total Number of Students Finish the Quiz: 10
--

TABLE IX (Continue)

Question Number	number of students who choose			
#1	0	1	8	1
#2	1	8	1	1
#3	1	0	1	8
#4	0	1	8	1
#5	2	4	4	0
#6	2	0	7	1
#7	7	0	2	1
#8	0	8	2	0
#9	1	1	8	0
#10	0	8	1	1

Summary:

Question #	Correct Percentage of the Students
#1	80
#2	80
#3	80
#4	80
#5	40
#6	70
#7	70
#8	80
#9	80
#10	80

Overall 10 questions, the average correct rate is 74

CHAPTER IV

DISCUSSION AND FUTURE WORK

Computer has been popularized in every area of human society, education certainly is one of the areas. The traditional education form, teachers give lectures by using black board, no longer is the sole form. Computer assisted instruction (CAI) plays an important role in education now.

This thesis studies a three-phased design model, applies this model in NRKS during the whole development circle. Also, it employs the storyboard concept to work out the best choice among the combination of all possible screen layout and sequence. These two mechanisms help to simplify instruction design, and to reduce design errors. The two way of views (system point of view and user point of view) and tree structure of topic organization methods make the design objective clear and easy to achieve. The individual report and the class report designed in this thesis are valuable in analyzing students' performance, and they can tell if the CAI lesson is appropriate to the target population.

Nature resource knowledge system (NRKS) is a CAI lesson that teaches students the knowledge about tree, forestry, urban, wildlife etc. NRKS employs pictures and music into instruction. The animation and window video can be successfully implemented in this package, both animation and video can play with music. Well designed animation and video application can be added in NRKS.

Due to the large number of graphics, this package needs a significant amount of hard disk space. The animation and video files take even more space. Therefore using CD as the storage device is considered in the future.

Detail information about this software can be obtained by contacting with Dr. G.W. Cuperus, Entomology department (127 NRC), Oklahoma State University, Stillwater, OK, 74078.

BIBLIOGRAPHY

- [Aarntzen, 93] Diana Aarntzen, *Audio in Courseware: Design Knowledge Issues*, ETTI 30, 4 pp. 354-363. 1993
- [Becker, 89] Becker, H. J., *United States administration of the 1989 IEA international survey of computers in education* (preliminary data, unpublished report). Baltimore, MD: Johns Hopkins University, Center for Social Organization of Schools. 1989
- [Brewer, 86] Brewer, B, *Compact disc interactive audio*. In Lambert, S. and Ropiequet, S. (eds), *CD-ROM: The new papyrus* (PP. 273-290). Microsoft Press, Redmond, VA, 1986.
- [Dick & Carey, 85] Dick, W., and L. Carey, *The Systematic Design of Instruction*(2nd ed.), 1985
- [Hannafin & Peck, 88] Michael J. Hannafin & Kyle L. Peck, *The Design, Development, and Evaluation of Instructional Software*, 1988
- [Hirschbuhl, 89] Hirschbuhl J. J, *CBT survey for computer based training directions. Interactive Learning International*, 5(2), pp.55-77. 1989.
- [House, 82] House, W. C., *Interactive Computer Graphics Systems*, Petrocelli Books, Inc, 1982.
- [Jannafin, 88] Michael J. hannafin, *The Design, Development, and Evaluation of Instructional software*, pp. 59-63. 1988.
- [Kumar, 94] David D. Kumar, *Hypermedia: a Tool for Alternative Assessment*. ETTI 31,1 pp. 59. 1994.
- [Lamb, 92] Annette C. Lamb, *Multimedia and the Teaching-Learning Process in Higher Education*, New Directions for teaching and Leering, no. 51, Fall 1992.
- [Park, 94] Ok-choon Park, *Dynamic Visual Displays in Media-based Instruction*, Educational Technology/April 1994.

- [Pritchard, Micceri, & Barrett, 89] Pitchard, W. H., Jr., Micceri, T., & Barrett, A. J. A review of computer-based training materials: Current state of the art (Instruction and interaction). *Educational technology*.
- [Roblyer, 88] Roblyer, M. D, Fundamental problems and principles of designing effective courseware. In D. H. Jonassen (Ed.), *Instructional designs for microcomputer courseware* (PP. 7-33). Hillsdale, NJ: Lawrence Erlbaum Associates. 1988.
- [Sloane, 89] Howard N. Sloane, *Evaluating educational software*. Prentice-Hall, Inc. 1989.
- [Steinberg, 91] Esther R. Steinberg, *Teaching Computers To Teach*, pp.1-11. 1991.
- [Steinberg, 91a] Steinberg, E. R., *Computer-assisted instruction: A synthesis of theory, practice, and technology*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- [Taylor, 88] Taylor, M. H, *Planning for video: A guide to making effective training videotapes*. Kogan Page, London, Nichols, New York. 1988.
- [Venezky & Osin, 91] Richard Venezky, Luis Osin, The intelligent design of computer assisted instruction, pp. 31-39. 1991.
- [Wishnietsky, 92] Dan H. Wishnietsky, *Hypermedia: The Integrated learning Environment*, pp. 7-23. 1992.

[^]
VITA

Xia Chen

Candidate for the Degree of

Master of Science

Thesis: A CAI SYSTEM FOR LEARNING NATURAL RESOURCE KNOWLEDGE
(NRKS)

Major Field: Computer Science

Biographical:

Personal Data: Born in Wenzhou, China, August 13, 1967, the daughter of Dexiao Chen and Ying Wang.

Education: Graduated from The First Middle School, Wenzhou, China, in June 1985; received Bachelor of Science Degree from Zhejiang University in July 1989; completed requirements for the Master of Science degree at Oklahoma State University in December, 1994.

Professional Experience: Assistant Engineer, West Lake Electronics Inc., August, 1989, to January 1992; Research Assistant, Department of Entomology, Oklahoma State University, January, 1994, to December 1994.