Technology Institute for Music Educators

Multimedia Authoring
(Teacher Manual)

by Kim Walls and David Sebald

Edited by Floyd Richmond

Copyright © 1999 Technology Institute for Music Educators
http://www.ti-me.org

These materials were made possible by a grant from NAMM (National Association of Music Merchants)
# Table of Contents

## Multimedia Authoring
- Description .................................................. 4
- Course Organization ........................................... 4
- Additional Information ........................................ 4
- Hardware Requirements ....................................... 5
- Software Requirements ....................................... 5

## Part 1 - Multimedia Presentation Software
- Introduction ................................................... 6
- Prerequisites ................................................... 6
- Objectives ....................................................... 6
  - Declarative Knowledge .................................... 6
  - Procedural Knowledge .................................... 6
- Assessment ..................................................... 6
- Course Topics ................................................ 7
  - Basic Operation ............................................ 7
  - Creating a Presentation .................................. 7
  - Incorporating Graphics .................................. 7
  - Incorporating Sounds .................................... 8
  - Individual Project Development ....................... 8
- Teacher Worksheets 1 - 5 .................................. 9 - 13

## Part 2 - Stack-Based Interactive Multimedia Authoring
- Introduction ................................................... 14
- Prerequisites ................................................... 14
- Description .................................................... 14
- Objectives ....................................................... 14
  - Declarative Knowledge .................................... 14
  - Procedural Knowledge .................................... 14
- Assessment ..................................................... 15
- Course Topics ................................................ 16
  - Hypermedia Objects ...................................... 16
  - Buttons and Actions ..................................... 16
  - Other Objects and Actions .............................. 16
  - Hypertext and Circular Stacks ......................... 17
  - Creating a Linear Stack .................................. 17
  - Creating a Branching Stack ............................. 18
  - Testing Functions ........................................ 18
- Design Guidelines ............................................ 18
- Curricular Integration and Copyright .................. 19
- Distributing Presentations ............................... 19
- Teacher Worksheets 1 - 10 ................................. 20 - 33
- Class Project Downloading and Setup Instructions. 34
- In-Class Projects ............................................ 35 - 47
### Part 3 - Time-Line Interactive Multimedia Authoring

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>48</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>48</td>
</tr>
<tr>
<td>Description</td>
<td>48</td>
</tr>
<tr>
<td>Objectives</td>
<td>48</td>
</tr>
<tr>
<td>Declarative Knowledge</td>
<td>48</td>
</tr>
<tr>
<td>Procedural Knowledge</td>
<td>49</td>
</tr>
<tr>
<td>Assessment</td>
<td>49</td>
</tr>
<tr>
<td>Course Topics</td>
<td>49</td>
</tr>
<tr>
<td>Multimedia and Director</td>
<td>49</td>
</tr>
<tr>
<td>Director Basics</td>
<td>50</td>
</tr>
<tr>
<td>Noninteractive Animation</td>
<td>51</td>
</tr>
<tr>
<td>Interactive Multimedia Product Design</td>
<td>51</td>
</tr>
<tr>
<td>Lingo and Interactivity</td>
<td>52</td>
</tr>
<tr>
<td>LINGO AND INTERACTIVITY 2</td>
<td>52</td>
</tr>
<tr>
<td>LINGO AND INTERACTIVITY 3</td>
<td>53</td>
</tr>
<tr>
<td>Delivery Considerations</td>
<td>53</td>
</tr>
<tr>
<td>EXTENDING DIRECTOR</td>
<td>55</td>
</tr>
<tr>
<td>Students and Time-Line Multimedia Authoring</td>
<td>55</td>
</tr>
<tr>
<td>Final Demo and Product Critique</td>
<td>55</td>
</tr>
<tr>
<td>Student Worksheets and Handouts</td>
<td>56-89</td>
</tr>
</tbody>
</table>

### Appendix 1 - Multimedia Authoring Programs

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix 1 - Multimedia Authoring Programs</td>
<td>90</td>
</tr>
</tbody>
</table>
Multimedia Authoring (TI:ME 2B)
David Sebald and Kim Walls, authors
Floyd Richmond, editor
last modified June 1, 1999

Copyright © 1999 by the Technology Institute for Music Educators

• Multimedia Presentation Software
• Stack-Based Interactive Multimedia Authoring
• Time-Line Interactive Multimedia Authoring

Description
Multimedia Authoring (TI:ME 2B) covers the creation of multimedia for the music classroom. Inservice teachers (ISTs) who complete this course will create and use multimedia presentations and interactive lessons which are appropriate for the K-12 music classroom. ISTs will incorporate digital multimedia files including text, graphics, video, sound, and music into presentations and will incorporate interactivity into music lessons. Software to be used in the class will include a presentation program and either a stack-based or time-line authoring program.

The format of the course will alternate presentations with class activities in which student progress is assessed and in which the material presented is reinforced. Evaluation will be through written homework, completion of class assignments, and the submission of a final project demonstrating techniques learned in class. This course meets one third of the coursework requirements for Level Two TI:ME Certification. Prerequisites for this course include completion of both TI:ME level 1 courses OR equivalent experience.

Course Organization
This course consists of three segments: multimedia presentation, stack-based multimedia, time-line multimedia. Students taking this course will complete the multimedia presentation segment and ONE of the other two segments as indicated by the instructor.

Additional Information
Multimedia Authoring (TI:ME 2B) may be offered as a two-credit or three-credit graduate course. Topics in ALL CAPS within the outline need not be included when it is offered for two graduate credits only. The instructor of this course must be approved by TI:ME and experienced in teaching technology to inservice music teachers. This course will serve a maximum of 16:1 teacher/student ratio. Additional students may be accommodated if additional workstations and teacher assistants are available. Each IST will need approximately 20 hours working alone at a workstation in order to complete class activities and the final project.
Hardware Requirements
A computer lab/classroom consisting of up to sixteen student and one teacher's computer/music workstation is the required minimum configuration. The classroom must have the ability to connect to the Internet, with direct LAN access a plus. A telephone connection and modem at the teacher's station is also required. Each participant must have access to a private workstation consisting of a multimedia-capable computer, MIDI keyboard, and audio and video support equipment. The teachers station should be connected to a projection device and all student and teaching stations should be connected to an audio playback system for class evaluation of each other's work. Hardware for creating and editing digital multimedia such as scanners, microphones, digital cameras (still pictures) and digital cameras (movies) should be available to students as needed.

Software Requirements
This course requires the following software:

- A "slide-show" presentation program
- A multimedia authoring program with either a stack-based or time-line organization.
- Programs for creating and editing text, graphics, sounds, and video.
- An Internet browser.

All software choices (or equivalent programs) should be available for Mac OS and Windows platforms. Numerous multimedia authoring programs can be used to complete the requirements of this course. This handbook presents screen shots and explanations of techniques which may seem to favor one program over another. TI:ME does not endorse or prefer any specific program but encourages instructors to use modern and effective multimedia authoring software. Appendix 1 provides a list of multimedia authoring and presentation programs
PART 1

Multimedia Presentation Software

Kimberly C. Walls

**Introduction**
The outline is designed for a 10 hour unit on multimedia presentation software within a 2 credit graduate workshop. Items in UPPERCASE can be added if the course if offered for 3 graduate credits. All students completing the TI:ME Multimedia Authoring course should complete the Multimedia Presentation Segment.

The primary objective of this section is to instruct ISTs in basic skills in using multimedia presentation software. The instructor will also provide ISTs with the information they need to plan educationally valid integration of multimedia presentations into their teaching and legal usage of media. In addition to satisfactory participation in class activities, a multimedia project is required for certification.

**Prerequisites**
Enrollees in this course should have at least basic computing skills at the level of word processing and familiarity with a graphic user interface (Windows or Macintosh OS). Concurrent or previous enrollment in a course emphasizing digital media acquisition and manipulation would also be helpful.

**Objectives**

**Declarative Knowledge**
The IST describes several ways to use slide-show presentations to enhance teaching and learning.
The IST identifies and describes copyright issues applicable to multimedia authoring.

**Procedural Knowledge**
The IST demonstrates basic skills in using a slide-show multimedia presentation program to produce classroom materials. The skills to be acquired include the following:

- Playing an existing presentation
- Creating an original presentation
- Adding text and graphics to a presentation
- Add audio and MIDI files to a presentation
- Adding animations and digital movies to a presentation
- Adding visual transitions to a presentation

**Assessment**
The IST completes the class worksheets on multimedia presentation software. The IST completes a multimedia "slide-show" presentation useful for the classroom. The project will be evaluated in terms of its professionalism, suitability, and creativity.
Multimedia Presentation Software

COURSE TOPICS

**Topic #1**
**Basic Operation**
The instructor will explain the concept of a multimedia presentation. The instructor will demonstrate basic operation of Multimedia presentation software: opening a presentation, playing a slide-show, using the standard toolbar, working with views. The instructor will review multimedia definitions and media types.

**Class Activities #1**
The IST will open an existing presentation and play a slide-show. The IST will explore the Standard toolbar and Views.

**Topic #2a**
**Creating a Presentation**
The instructor will demonstrate creating a presentation: using templates and AutoWizards, choosing color schemes, using slide masters, entering text, changing text styles, bullets, slide transitions, checking spelling, changing the background.

**Class Activities #2a**
The IST will use a template or AutoWizard to create a text-based presentation.

**TOPIC #2B**
The instructor will demonstrate CHANGING SLIDE MASTERS, ADDING HEADERS AND FOOTERS, AND ADDING HYPERLINKS.

**CLASS ACTIVITIES #2B**
The IST will change a slide master, add headers and footers, and add a hyperlink.

**Topic #3a**
**Incorporating Graphics**
The instructor will discuss elements of visual design and demonstrate using clip art, drawing tools, and external graphics files.

**Class Activities #3a**
The IST will use a template or AutoWizard to create a text-based presentation.
TOPIC #3B
THE INSTRUCTOR WILL DEMONSTRATE ADDING ANIMATION EFFECTS AND ACTION BUTTONS.

CLASS ACTIVITIES #3B
THE IST WILL ADD ANIMATION EFFECTS AND ACTION BUTTONS.

---

Topic #4a
Incorporating Sounds

The instructor will discuss the benefits of incorporating sounds and video into a multimedia presentation and demonstrate adding audio CD, external sound files, and video into a presentation.

Class Activities #4a
The IST will add audio CD, external sound files, and video into a presentation.

TOPIC #4B
THE INSTRUCTOR WILL DEMONSTRATE RECORDING NEW SOUND CLIPS AND NARRATING SLIDES.

CLASS ACTIVITIES #4B
THE IST WILL RECORD NEW SOUND CLIPS AND NARRATE SLIDES.

---

Topic #5a
Individual Project Development

The instructor will discuss designing a slide-show presentation for use in a music class and how to integrate the show into a class.

Class Activities #5a
The IST will design and create a slide-show presentation for use in a music class.

TOPIC #5B
THE INSTRUCTOR WILL DEMONSTRATE SAVING A SLIDE-SHOW FOR INTERNET PRESENTATION.

CLASS ACTIVITIES #5B
THE IST WILL SAVE A SLIDE-SHOW FOR INTERNET PRESENTATION.
1. Explain the concept of a slide-show multimedia presentation.

   [A slide-show is a multimedia presentation that generally is linear (screens are shown sequentially forwards and/or backwards), but may offer branching options too. A slide-show is not scriptable.]

2. How does one open a multimedia presentation?

   [Either with the File menu or the Standard toolbar Open icon.]

3. How does one play a slide-show and what are some of the playback options?

   [Use the Slide-Show option under the Views menu to play from the first slide or use View Show under the Slide-Show menu. Use the Slide-Show view button to start with the currently displayed slide. Playback options are listed under Slide-Show, Set Up Show...]

4. How are the following views typically used?

   a. Slide View
      [To work on one slide, make changes to visual design, especially backgrounds and margins.]
   
   b. Outline View
      [To work on the text content of the slide-show and the structure of the content.]
   
   c. Slide Sorter View
      [To rearrange the order of slides, or see a visual overview of slides.]
   
   d. Notes Page View
      [To make speaker notes and/or handouts.]
   
   e. Slide-Show View
      [To run a slide-show electronically.]

5. Describe the multimedia elements listed below.

   a. Text [Text equals words. Although multimedia is a graphically rich environment, much of its meaning would be lost without the words which give specific instruction and content.]
   b. Graphics - [Graphics are pictures. Pictures are used to illustrate points or to add beauty.]
   c. Sounds - [Sounds in multimedia projects are primarily digital recordings of sounds.]
   d. MIDI files - [MIDI files are the files generated by sequencing or notation software. These files contain instructions for playing music on an external device such as a synthesizer or sound card. MIDI files are quite compact but they do not contain digital recordings of specific sounds.]
   e. Movies - [Movies are digital recordings much like video tape.]
   f. Interactive Links - [Interactive links are connections from one document to another. When the user clicks on a link, the computer displays the indicated part of the document.]
Multimedia Presentation Software

TEACHER WORKSHEET 2
Creating a Presentation

1. How/why are templates and AutoWizards used?
   [Open them and fill place holders with text or other media. They provide a consistent style for a presentation.]

2. How is text entered?
   [Use outline view to enter text. Use Slide view to change positioning.]

3. What are slide transitions? How does one add them?
   [Visual effects between slides.]

4. How does one add/change bullets?
   [Select text and click on Bullets button on Formatting toolbar.]

5. How does one change which slide master is assigned to a slide?
   [Format menu --> Slide Layout.]

6. How does one check spelling?
   [Tools menu --> Spelling]

7. How does one change the background of a slide?
   [Select slides, Format menu --> Background, drop down list in Background Fill box.]

8. WHY/HOW ARE SLIDE MASTERS MANIPULATED?
   [VIEW MENU --> MASTER. CHANGE POSITIONING AND STYLE OF PLACEHOLDERS AND TEXT.]

9. WHY/HOW ADD HEADERS AND FOOTERS?
   [TO PROVIDE CONTINUITY TO SLIDE-SHOW. VIEW --> HEADER AND FOOTER...]

10. WHY/HOW ADD HYPERLINKS?
    [TO BRANCH TO DIFFERENT SLIDES, OR AN INTERNET SITE, OR START ANOTHER PROGRAM. USE HYPERLINK BUTTON ON STANDARD TOOLBAR.]
Multimedia Presentation Software

TEACHER WORKSHEET 3

Incorporating graphics

1. Why would you use graphics in a multimedia presentation?
   [To enhance meaning and add information. "A picture is worth a thousand words."]

2. Describe functions of the different graphics functions:
   a. The Clip Gallery
      [To insert PowerPoint clip art.]
   b. Insert-->Picture-->From File
      [To insert external graphics files.]
   c. Word Art Toolbar
      [To create text effects that are graphics, not text objects.]

3. How does one use the Preset Animation effects?
   [While in Slide View, Slide-Show-->Preset Animation]

4. HOW DOES ONE ACCESS THE CUSTOM ANIMATION EFFECTS?
   [WHILE IN SLIDE VIEW, SLIDE-SHOW-->CUSTOM ANIMATION]

5. HOW DOES ONE ADD ACTION BUTTONS?
   [IN SLIDE VIEW, SLIDE-SHOW-->ACTION BUTTONS]
Multimedia Presentation Software

TEACHER WORKSHEET 4
Incorporating sounds

1. List some ways to use sounds in a multimedia presentation.
   [Present musical examples, narrate show, background music.]

2. How does one add audio CD clips to a slide-show?
   [In Side View, Insert-->Movies and Sounds-->Play CD Audio Track, set start and stop times. It will play during that slide only.]

3. How does one have an audio CD play through more than one slide?
   [While in Slide View, Slide-Show-->Custom Animation]

4. How does one add external sound files to a slide-show?
   [Insert-->Movies and Sounds-->Sound from File]

5. How does one add video clips to a slide-show?
   [Insert-->Movies and Sounds-->]

6. HOW DOES ONE RECORD NEW SOUND CLIPS?
   [INSERT-->MOVIES AND SOUND-->RECORD SOUND]

7. HOW DOES ONE ADD SLIDE NARRATION?
   [FINISH AND SAVE PRESENTATION. SLIDE-SHOW-->RECORD NARRATION LINK NARRATION IN. ADVANCE SLIDES AND RECORD.]
1. List some issues to keep in mind when designing a slide-show for use in a music class.

2. List some ways a slide-show presentation can be used in the classroom.

3. WHAT ARE SOME OPTIONS FOR PRESENTING A SLIDE-SHOW THROUGH THE INTERNET?
PART 2
Stack-Based Interactive Multimedia Authoring

Kimberly C. Walls

Introduction
The outline is designed for a 20 hour unit on HyperStudio (or other stack-based authoring program such as HyperCard, SuperCard, ToolBook) within a 2 credit graduate workshop. Items in UPPERCASE can be added if the course if offered for 3 graduate credits.

Prerequisites
Enrollees in this course should have at least basic computing skills at the level of word processing and familiarity with a graphic user interface (Windows or Macintosh OS). Concurrent or previous enrollment in a course emphasizing digital media acquisition and manipulation would also be helpful.

Description
The primary objective Multimedia Authoring is to impart practical skills and knowledge to in service teachers (ISTs) to allow them to integrate technology effectively into teaching and learning. The purpose of this section of Multimedia Authoring is to instruct ISTs in basic skills in using a stack-based authoring program. The instructor will also provide ISTs with the information they need to plan educationally valid integration of multimedia authoring and productions into their teaching and legal usage of media. The format of this section alternates presentations with class activities, many of which serve to assess student progress. In addition to satisfactory participation in class activities, a multimedia project plan is required for certification.

Objectives
Declarative Knowledge
The IST describes the differences among and advantages of linear, branching, and circular multimedia programs.
The IST identifies and describes copyright issues applicable to multimedia authoring.
The IST describes several ways to use stack-style interactive multimedia to enhance teaching and learning.

Procedural Knowledge
The IST demonstrates basic skills in using a stack-based multimedia authoring package to produce interactive programs. The skills to be acquired include the following:
- Adding existing cards, text objects, and graphic objects to a stack.
- Creating buttons and objects to play sounds, navigate to other cards, play digital movies, play an animation, hide/show objects, play CD excerpts, play MIDI files, and play visual transitions.
- Create hypertext links.
- Create a linear presentation, a branching presentation, and a quiz.
- Format presentations for an Internet browser.
- DIGITIZE STILL IMAGES FROM WITHIN THE AUTHORING PROGRAM.
- ADD SCRIPTS TO BUTTONS.
- ADD URL LINKS TO OBJECTS.
- ADD TIMING FUNCTIONS.
- USE GROUPED CARDS
- PRINT QUIZ RESULTS AND WRITE QUIZ RESULTS TO EXTERNAL FILES
- ASSESS PRODUCTS.
- POST INTERNET FILES TO A SERVER.

Assessment
The IST completes the class worksheets on stack-based multimedia authoring software.
The IST completes a stack-based interactive multimedia lesson useful for the classroom. The project
will be evaluated in terms of its professionalism, suitability, and creativity.

Downloading and setup instructions for class projects are found on page 30.
Stack-Based Interactive Multimedia Authoring
COURSE TOPICS

**Topic #1a**
**Hypermedia Objects**
The instructor will explain the stack metaphor and the role of objects in hypermedia by displaying a demonstration hypermedia stack. (Use any of the flashy demo stacks that come with HyperStudio.) The instructor will demonstrate how to navigate through a stack.

**Class Activities #1a**
ISTS will open an existing stack (PROJECT1.STK) and add text to a text object, angle the style of text, change a background, add existing cards, add clip art, paint a background, and add graphic objects to it.

**TOPIC #1B**
**PRESENTATION 1B: THE INSTRUCTOR WILL DEMONSTRATE DIGITIZING STILL IMAGES WITH A SCANNER OR VIDEO SOURCE FROM WITHIN THE AUTHORING PROGRAM.**

**CLASS ACTIVITIES #1B**
ISTS WILL DIGITIZE STILL IMAGES WITH A SCANNER OR VIDEO SOURCE FROM WITHIN THE AUTHORING PROGRAM.

---

**Topic #2a**
**Buttons and Actions**
The instructor will explain the role of actions and linking in hypermedia and demonstrate how to create various types of buttons.

**Class Activities #2a**
ISTS will add various types of buttons to PROJECT2.STK during which play a sound, navigate to another card, play a digital movie, play an path-based animation, and hide/show objects.

**TOPIC #2B**
**THE INSTRUCTOR WILL DEMONSTRATE HOW TO ADD A CELL ANIMATION (CELL.STK).**

**CLASS ACTIVITIES #2B**
ISTS WILL CREATE A CELL ANIMATION

---

**Topic #3a**
**Other Objects and Actions**
The instructor will demonstrate how to add actions to other objects.

**Class Activities #3a**
ISTS will add the following actions to graphic objects, text objects, cards, and the stack: play a CD excerpt, play sound files, play a MIDI file, and play visual transitions.
TOPIC #3B
THE INSTRUCTOR WILL DEMONSTRATE ADDING NAVIGATION, HIDE/EASE, AND TOOT SCRIPTS.

CLASS ACTIVITIES #3B
ISTS WILL ADD SCRIPTS TO OBJECTS IN KARAOKE.STK.

---

**Topic #4a**

**Hypertext and Circular Stacks**
The instructor will demonstrate creating text objects and hypertext links. The instructor will describe circular stacks and using the find command to find cards.

Class Activities #4a
ISTS will add text objects, hypertext links, group cards, a button with a menu command, and a title card to PROJECT4.STK.

**TOPIC #4B**
THE INSTRUCTOR WILL DEMONSTRATE WRITING SCRIPTS TO PLAY SOUNDFILE EMBEDDED WITHIN A STACK (SOUNDER.STK).

CLASS ACTIVITIES #4B
ISTS WILL WRITE SCRIPTS TO PLAY SOUNDFILE EMBEDDED WITHIN A STACK (SOUNDER.STK).

---

**Topic #5a**

**Creating a Linear Stack**
The instructor will describe the design of a timed linear presentation.

Class Activities #5a
ISTS will create a multimedia listening companion (PROJECT5.STK) using grouped buttons, automatic timers, and CDPlay NBA.

**TOPIC #5B**
THE INSTRUCTOR WILL DEMONSTRATE SCRIPTING TIMING FUNCTIONS TO A LINEAR STACK (PROJECT5.STK).

CLASS ACTIVITIES #5B
ISTS WILL ADD SCRIPTED TIMING FUNCTIONS TO A LINEAR STACK (PROJECT5.STK).
**Topic #6a**  
*Creating a Branching Stack*  
The instructor will describe the design of a branching presentation.

Class Activities #6a  
ISTs will create a branching presentation based upon a musical composition (PROJECT6.STK).

**Topic #6B**  
*The Instructor Will Demonstrate Scripts for User Input and Text Object Output.*

Class Activities #6B  
ISTS will write scripts for user input and text object output.

**Topic #7a**  
*Testing Functions*  
The instructor will demonstrate adding testing functions to a branching presentation.

Class Activities #7a  
ISTs will add a quiz to a branching presentation based upon a musical composition (PROJECT7.STK).

**Topic #7B**  
*The Instructor Will Explain How to Add Scripts to Print Present Quiz Results or Write Quiz Results to External Files.*

Class Activities #7B  
ISTS will add scripts to print present quiz results or write quiz results to external files.

**Topic #8a**  
*Design Guidelines*  
The instructor will lead a discussion of designing/storyboarding/flow-charting hypermedia productions.

Class Activities #8a  
ISTs will discuss designing a hypermedia product by following questions on design worksheet.

**Topic #8B**  
*The Instructor Will Discuss Assessment of Pilot Software.*

Class Activities #8B  
ISTS will design an assessment of the designed product.
**Topic #9a**

**Curricular Integration and Copyright**
Issues The instructor will lead a discussion of curricular integration of hypermedia and copyright issues.

Class Activities #9a
ISTS will describe curricular integration of hypermedia and copyright issues on curriculum/copyright worksheet. They will add “credits” and a “bibliography” to PROJECT6.STK using the Ready Made Cards.

**TOPIC #9B**
THE INSTRUCTOR WILL DEMONSTRATE THE USE OF COMPUTER PROJECTION DEVICES AND AMPLIFIED SPEAKERS FOR MULTIMEDIA PRESENTATION.

**CLASS ACTIVITIES #9B**
ISTS WILL DISPLAY ORIGINAL STACKS USING COMPUTER PROJECTION DEVICES AND AMPLIFIED SPEAKERS.

---

**Topic #10a**

**Distributing Presentations**
The instructor will demonstrate how to format presentations for the Internet and other operating systems.

Class Activities #10a
ISTS will save branching presentation for the Internet and other operating systems.

**TOPIC #10B**
THE INSTRUCTOR WILL DEMONSTRATE HOW TO POST INTERNET FILES TO A SERVER.

**CLASS ACTIVITIES #10B**
ISTS WILL POST INTERNET FILES TO A SERVER.
Stack-Based Interactive Multimedia Authoring
TEACHER WORKSHEET 1
Hypermedia Objects

1. What is a "stack" metaphor? A HyperStudio project is a stack. A stack of what?
   [Each screen that is displayed in a "stack" metaphor is an
   object called a "card." A collection of "cards" is a "stack" of "cards."
   A stack is also an object.]

2. What are some types of objects found in a hypermedia project?
   [Cards, text objects, graphic objects, buttons.]

3. How does one add text to or change text in an existing text object?
   [Make sure the Browse Tool is selected, click in an existing text object
   and type.]

4. How does one add a graphic object to a stack?
   [Choose Add a Graphic Object from the Objects menu.]

5. What is the difference between graphic objects and background or clip art?
   [Background graphics and clip art are part of the card and may be edited
   with the Paint tools. Graphic objects "float" on top of the card.
   Backgrounds are identical on grouped cards.]

6. How does one add a card to a HyperStudio stack?
   [Use the New Card, Ready Made Cards, or Copy Card and Paste Card
   commands under the Edit menu.]

7. HOW DOES ONE DIGITIZE IMAGES FROM WITHIN THE HYPERSTUDIO PROGRAM?
   [USE THE ADD CLIP ART COMMAND UNDER THE FILE MENU AND SELECT THE
   CAMERA OR OTHER VIDEO DEVICE THAT IS ATTACHED TO THE COMPUTER.
   NOTE: THIS SHOULD BE DONE ON A BLANK NEW CARD IF THE CLIP ART IS TO
   BE CONVERTED TO A GRAPHIC OBJECT INSTEAD OF A BACKGROUND IMAGE.]
1. What are buttons?
   [Buttons are areas of the card which perform an action when the cursor is placed on the area and the user clicks the mouse button.]

2. What are some typical actions associated with buttons?
   [Display a different card or stack, start another program, play a sound file, play a digital movie, display another object, run a script.]

3. How does one create a button?
   [Select Add a button from the Objects menu. Select the type, shape, name, icon, position and action.]

4. HOW DOES ONE CREATE A CELL ANIMATION? [CREATE FRAMES OF ANIMATION, ONE PER CARD, THEN EXPORT CARD SCREENS]
1. How does one add actions to existing graphic or text objects?  
   [Use the Arrow tool and double-click on the object to edit it.]

2. How does one add actions to cards or a stack?  
   [Under the Objects menu, choose About this Card or About this Stack and define the Actions]

3. How does one play an excerpt from a CD?  
   [Create or edit an object's action to be a New Button Action. Choose the CDPlay new button action and click on Use this NBA. The CD controller will allow you to set up the excerpt to be played.]

4. How does one play a MIDI file?  
   [The same as playing a digital movie, just select a .mid file instead of a movie. HyperStudio will convert it to a QuickTime MIDI movie.]

5. How does one add visual transitions?  
   [Either create or edit an object so that its action is to go to another card or stack. After selecting which location, choose a transition.]

6. HOW DOES ONE ADD A SCRIPT TO A BUTTON?  
   [CHOOSE USE HYPERLOGO FROM THE ACTIONS DIALOG OR COMMAND-CLICK A BUTTON.]

SCRIPTING NOTES

Navigation commands: MOVENEXT, MOVEPREV, MOVETOFIRST, MOVETOLAST, MOVETOCARD Cardname

Hiding/Showing Objects on a card to create an animation:
SHOWITEM [] Objectname Objecttype
WAIT NumberOfSixtiethsOfSeconds
HIDEITEM [] Objectname Objecttype

Playing pitches with tooter:
TOOT MIDInote NumberOfSixtiethsOfSeconds
1. How does one create a new text object?  
   [Choose Add a Text Object from the Objects menu.]

2. What is hypertext?  
   [A portion of text in a text object which has an action assigned to it.  
   It is an object too.]

3. How does one create a hypertext link?  
   [Highlight the text using the browse tool, choose Hypertext Links from  
   the Objects menu, click on Add link, click on Action and add the action,  
   then click on Done.]  

4. What is a circular stack and why would one choose to design a  
   circular stack?  
   [In a circular stack, the cards are arranged so that it does not matter  
   which order the cards are viewed. The design is useful for collections  
   of information in which each card has similar objects, such as various  
   information about music students, one student per card. In this design,  
   it is important to be able to search/find information, and/or create reports.]

5. How does one find text in a stack? Which text may be found?  
   [Use the Find Text command under the Move menu. Text in text objects may  
   be found.]

6. How does one add a button action to find text?  
   [Use the MenuChooser NBA and type Move Find Text1]

7. What is the function of grouped cards? [Grouped cards have the same  
   background and card actions. Objects on the grouped cards that have been  
   grouped also appear on each card.]

SCRIPTING NOTES  
See Chapter 6 of Exploring HyperLogo Tutorial by Bill Lynn, available  
through www.hyperstudio.com
1. What is a linear stack and why would one choose to design a linear stack?
   [In a linear stack, the cards are arranged so that the cards are viewed in a given order. The design is useful for presenting information in a step-wise fashion, such as images of the proper order of assembling a music instrument. The student may view each card at his own pace, but the order of cards is set. Timing functions can be useful for presenting images coordinated with a musical recording.]

2. How does one add timing functions to a linear stack?
   [For any object, use the automatic timer option in the Actions dialog.]

SCRIPTING NOTES
Use MOVENEXT, and WAIT to create a button that presents each card for a specified time. Use a REPEAT loop and IF BUTTONP to set up a WAIT that can be interrupted by a mouse click. (Delete the invisible timing buttons on each card.) See Chapter 2 of Exploring HyperLogo Tutorial by Bill Lynn, available through www.hyperstudio.com
Stack-Based Interactive Multimedia Authoring
TEACHER WORKSHEET 6
Creating a Branching Stack

1. What is a branching stack and why would one choose to design a branching stack?
   [A branching stack has cards that present navigation choices to the user. The choices are displayed as buttons or text options. Navigating The Stack is similar to using an index. The benefits of a branching presentation is that the user can choose which topics to view and when to view them.]

SCRIPTING NOTES
Use GETFIELDTEXT, SETFIELDTEXT, and WORD to ask the user for input, manipulate it and display a reply in PROJECT6.STK. (Need input & output text objects and an input button.) See "Creating a Review Stack" in Chapter 4 of Exploring HyperLogo Tutorial by Bill Lynn, available through www.hyperstudio.com
1. How does one add testing functions to a card? [Create the buttons or objects on the card and assign testing functions from the Actions screen. There MUST be at least one correct answer on each card for proper operation.]

2. EXAMINE THE BUTTON SCRIPTS IN QUIZBUTS.STK AND DESCRIBE HOW THEY OPERATE. [THEY USE THE GETCURRENTSCORE CALLBACK TO DERIVE THE USER NAME AND OVERALL TEST SCORES. ALSO, A TEXT FILE IS OPENED FOR READING AND INPUT INTO A TEXT OBJECT.]

3. EXAMINE THE FOLLOWING SCRIPT WHICH WILL PROMPT THE USER FOR AN ANSWER WHICH IS WRITTEN TO A FILE. HOW COULD THE SCRIPT BE EDITED TO WRITE TEST RESULTS TO A FILE AS A USER PROGRESSES THROUGH A STACK?

;close any open files
CLOSEALL
;name the file to write to
MAKE "resultfile "demoresults
;ask the dialog question
SETRWPROMPT "Write your question here?"
Make "TheAnswer READWORD
;open file for reading
OPEN :resultfile
PRINT :TheAnswer
;stop writing to file
SETWRITE []
;close files
CLOSEALL

[THE SCRIPT COULD BE CALLED WHEN THE USER CLICKS A RESPONSE BUTTON OR GOES TO ANOTHER CARD. OTHER COMMANDS COULD BE USED TO WRITE THE USERNAME AND TIME OR OTHER DATA TO THE FILE.]
Describe the process of designing/storyboarding/flow-charting hypermedia productions.
GUIDELINES FOR EFFECTIVE MULTIMEDIA DESIGN

by David Sebald

These guidelines have been gathered over a period of years from many resources on instructional design. Some concepts are expressed several times in different words, and some may appear to conflict. Still, each guideline has a logical justification in some context and when applied appropriately will result in a more effective multimedia project.

Rules for Efficient Production

Analyze the need:
- What is the purpose?
- Who is the intended user?
- What is the intended platform?

Research the topic thoroughly:
- What is the course content?
- What are the major topics/concepts?

Establish goals & objectives:
- What specific concepts, skills, attitudes should the user acquire through the instruction?
- How can these outcomes be measured?

Sequence instructional tasks:
- Write a flowchart on paper.

Gather all media.

Decide on appropriate production tools.

Create the computer program.

Package the product
- Produce accompanying documents.
- Create covers.

Implement the product.

Evaluate & revise the product.

Visual Interface Rules

Use Contrast to make things easy to distinguish and keep separated.
- Allow "frame" space around the screen elements.
- Distinguish different kinds of information visually.
- Leave lots of room around graphics.

Use Repetition to keep continuity both on a page and throughout a project.
- Reproduce the screen exactly in each frame.
- Keep continuity of screen elements (identity/size/placement) from frame to frame.
- Keep element locations constant.

Use Alignment to make screens sections aesthetically pleasing and to lead the viewer's eye appropriately.
- Titles should be at top and to the left.
- Start frame in the upper left.

Use Proximity to group related items together.
- Controls and options should be at bottom.

Use Balance to keep make the screen seem economical and aesthetically pleasing.
- Center the display.
- Use rule of thirds to place objects of importance.
- Weight the screen equally.

Use Emphasis to bring the most important things to the viewers attention.
- Omit extraneous detail.
- Keep readings short and logically segmented.
- Use a 24 line screen.
- Indicate topic and concept clearly on every frame.
- Keep relationships to all information clear.

Use Logic to put things where they make the most sense.
- Form should follow function.

Use Color for psychological reinforcement.
- Don't use blue on black.
- Don't use too many colors.
- 4 - 6 colors is best for an interface.
- Blue is always a safe background color.

Use Dynamics to imply feeling and for psychological reinforcement.
- Consider cultural values in symbol and color choice.

Keep the message more important than the medium.

Keep the interface clean (don't let effects get in the way of clarity.)
- Be economical in the use of screen gadgets and controls.
- Make sure that the viewer grasps meaning fast and easily.
- Don't clutter visuals with too much information.
Make symbols obvious, attractive, simple, informative, distinct, concrete, familiar, legible, and few in number.
Use "normal" photographs to illustrate concept, not photos that call attention away from it.
Use animation for dynamic concepts.
Keep screens orderly.
Use easily identifiable navigation tools and icons.

The Four Rules of Screen Design

Keep it consistent.
Keep it uncluttered.
Highlight important ideas.
Keep it readable.

Keep screen consistent:
Type size and font
Element location
Color
Control location and identity
Design and mood
Use lots of blank spaces

Rules for text:
Keep text large enough to read easily.
Don't stack text.
Use serifed fonts for text to be read.
Use sans-serifed fonts for headlines.
Avoid stylized fonts except as symbols.
Limit the sizes, styles, and colors to 3.
Don't use too much text to be read quickly in any text block.
Keep text as short as possible.
Use left justification in text blocks rather than full justification.
Keep text readable.
Grade level.
Wording.
Content.
Context.
Sequence.
Style.

Highlight important concepts:
Use plain English.
Use lots of blank spaces.
Don't duplicate aural narration with visual text.

Rules for user interaction:
Let the user control action as much as possible
Motivate the user.
Allow the user to control progression and navigation.
Allow for self-pacing but not procrastination.
All frames should contain an interactive component.
Keep the user oriented.
Frames can be multipurpose (present information & encourage manipulation).

General rules for effective instruction:

B.F. Skinner's 4 Basic Principles of instruction:
Formulate clear objectives.
Use substeps in large units.
Allow for individual progress.
Build new concepts on previously learned material.

Other Ideas:
Engage the user's attention. (enhances memory).
Present a question before presenting information. (doubles knowledge retention).
Clearly connect new information to previously acquired knowledge.
Keep information flowing (to prevent boredom).
Provide mnemonics and iconic or pictorial aids.
Ask the user to reconstruct steps leading to newly acquired concepts.
Use varied repetition & practice to cement important concepts.
Reward correct responses.
Do not use clever, sarcastic, or harsh punishments for wrong answers.
Keep the environment and problems as realistic as possible.
Keep the structure open and understandable.
Keep objectives clear and meaningful.
Keep instruction clearly sequenced, progressive, and in easily digested steps.
Avoid leaps of logic.
Don't bore the user.
Learning should be fun: Stimulating, motivating, captivating, engaging, rewarding.
Aim to the level of the user: Age, socio-economic group, familiarity with topic.
Rules for Physical Environment
Remember that reading text on paper is more efficient than reading text on screen.
Keep screen brighter than ambient light.
Keep screen contrast high.
Eliminate flicker.
Keep a center of acuity.
Eliminate head/neck strain.
Design for a 16” eye to screen distance.
Multimedia authoring, the National Standards and the TI:ME Strategies for Music Education

1. Which national standard(s) can be addressed using multimedia authoring software?

______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________

2. What specific ways could multimedia authoring be used to address these standards?

______________________________________________________________________
______________________________________________________________________
______________________________________________________________________

MENC Standards:
1. Singing, alone and with others, a varied repertoire of music.
2. Performing on instruments, alone and with others, a varied repertoire of music.
3. Improvising melodies, harmonies, and accompaniments.
5. Reading and notating music.
6. Listening to, analyzing and describing music.
7. Evaluating music and music performances.
8. Understanding relationships between music, the other arts, and disciplines outside the arts.
9. Understanding Music in Relation to History and Culture.

Now review the Teaching Strategies listed in the TI:ME technology strategies document. See Appendix A of the Technology Strategies for Music Education. Describe one or more ways to use multimedia authoring in your teaching position: [TIME INSTRUCTORS MAY PHOTOCOPY AND DISTRIBUTE APPENDIX A IF NECESSARY]

TI:ME Tech. Strategy # Teaching Application

___
___
___
___

List three ways that multimedia authoring could be used in your teaching position:

1. _______________________________________________________________________
2. _______________________________________________________________________
3. _______________________________________________________________________
Those who download files from the Internet and who create multimedia presentations and lessons should understand the copyright law. The right of creative people to control the display and circulation of their work is assured by law. As a general guideline, however, a copyright notice must be displayed on copyrighted material. If no notice is displayed, the material may be used but out of courtesy, permission should still be requested. If a copyright notice is displayed, then permission for using the material must be obtained from the copyright owner before using the material. On the Internet copyright permissions are easily obtained using e-mail. All materials not created by the author should contain a complete citation giving credit to the source.

For further information, refer to:

The United States Copyright Law - A Guide for Music Educators

http://www.menc.org/copy/copyr.html

Comments:
1. What are some aspects of stack design which should be considered when creating a stack for Web distribution?
[Screen size, numbers of colors, sound compression, and external QuickTime files. See Tips for Creating Web-hosted HyperStudio Projects at http://www.hyperstudio.com for more information.]

2. What are some aspects of stack design which should be considered when creating a stack for cross-platform distribution?
[Save stacks created on a Mac with a DOS-style name ending in .STK on a DOS disk to play on a Windows computer. Don't use spaces in the file name for Unix compatibility on the web. Stacks created with Windows can be read by a Mac. QuickTime movies should be flattened to be cross platform. Remember to include the movie files (external to stack). 256 colors will work best on Windows. External sounds should be .WAV files. AVI files should be converted to QuickTime for Mac playback. Use TrueType fonts which are supported on each platform or see the font conversion chart at http://www.hyperstudio.com.]

3. Which programs are used to play HyperStudio stacks when there is no copy of the HyperStudio on the computer? [Either the HyperStudio player or the HyperStudio plugin for browsers (see http://www.hyperstudio.com to download) is required to play a stack.]

4. What is the Export Web Page Extra for? [It creates an html file that automatically displays a HyperStudio stack.]

5. What are some ways HyperStudio could be used within web pages? [To add animation or user interaction in a portion of a web page. To control a CD from the web.]

6. NAME SOME GUIDELINES FOR POSTING STACKS ON A SERVER. [MAKE SURE THAT FILES ARE UPLOADED AS RAW DATA AND THAT A MIME TYPE HAS BEEN DEFINED. SEE http://www.hyperstudio.com FOR MORE INFORMATION.]
Stack-Based Interactive Multimedia Authoring
In-class Projects

**Downloading and Setup Instructions**

1. Download each of the following files as raw data files (about 4.2 MB):
   - PROJECT1.STK
   - PROJECT2.STK
   - PROJECT3.STK
   - PROJECT4.STK
   - PROJECT5.STK
   - PROJECT6.STK
   - PROJECT7.STK
   - PROJECT8.STK
   - QUIZBUTS.STK
   - QUIZBUTS.STK
   - KARAOKE.STK
   - SOUNDER.STK
2. Create a folder named TI-MEART and download these files into that folder:
   - MUSIC.PIC
   - TROMBONE.PIC
   - MARYTBN.MOV
3. Create a folder named TIMESNDS and download these files into that folder:
   - JOY.MID (MIDI file)
4. Download this file and put it in the "Ready made cards" folder within the HyperStudio folder: TI-MECard
Instructions
There are two buttons in the upper right section of the card. The "?" button will hide or display this text object (the one you are reading now). Click on underlined words for more information about a topic.

Before continuing with this project, go to the Edit menu and select Preferences. Put a check in the box beside "Show the card number with stack name" by clicking in the box. Also place a check in the box beside "I'm an experienced HyperStudio user."

If some of the options mentioned in these projects do not work as expected, go to Preferences and make sure that the boxes are checked.

There are some text objects on this card. One locked read-only text object contains the name of the stack. (It is covered by the object you are now reading. Click on the question mark button to hide this field.)

The text object at the bottom of the card has no text in it and it is unlocked. You may type in it.

Select the browse tool from the Tools palette and click in the text object below. Then type your name and today's date. You may highlight text and use the "Text Style" option under the Options menu to change the font, size, and style of the text. Use the "Set Text Color" option under the Options menu or use the palette under the Colors menu to change the color of text.

Click the Next Card button to continue.

Instructions - Card 2
(Adding a background and a graphic object to this card and adding a card to this stack.)

To change the color of this card's background, choose "Erase Background" from the Edit menu and select a color from the color picker.

Now add a graphic object to be displayed in front of the background. Choose "Add a Graphic Object" from the Objects menu.

Navigate to the "TI-MEART" folder and select "MUSIC.PIC". In the Graphic Objects screen, use either the Selector Tool, the Circular Selector, or the Lasso to select a musical symbol then click on "OK". While the red flashing dashes are surrounding the graphic object, you may drag it around the screen until it is in the desired location. Click outside of the graphic. Click on "OK" in the Graphic Appearance screen.

Add a new card to this stack by selecting "Ready Made Cards..." under the Edit menu and "TI-MECard". Follow the instructions on that card.
Instructions - Card 1

(Adding Buttons)

Go to the Objects menu and select Add a Button. At the Button Appearance dialog box, select the shaded rectangular button type, name the button Next Card, check "Show Name", "Highlight", and "Show Icon". In the Icons screen, select the right-pointing hand. Click the OK button, then click OK again. Drag the button so that it is the desired shape and in the desired location, then click on the card. In the Actions dialog, under Places to Go, select "Next Card".

Follow the same procedure but choose options to create a Previous Card button with a left-pointing hand icon which navigates to the previous card.

Next, add some invisible buttons to display the parts of the trombone covered by the buttons. Under the Objects menu select Add a Button then select the invisible rectangular button type. Click on Features. In the Features dialog, check "No Click" so that the button action will occur whenever the cursor is over the button (no click required) and click OK. Stretch the button to cover the trombone slide. Click on the card.

In the Actions dialog, select New Button Actions. Scroll to the HideShow NBA and click on Use this NBA. In the "Enter the name of the object" field, type slidetext, select "text object", and select "show". Click OK, OK, and Done.

Now if you have selected the browse tool, when the cursor is over the slide, the text field will appear.

Follow the same procedure to place an invisible button over the bell which displays the hidden text field named belltext and to place an invisible button over the tuning slide which displays the hidden text field named tuningtext.

Go to the next card using the button you created earlier.

Instructions - Card 2

(Adding Buttons)

Now add a visible rectangular button named Glissando. Make sure that the name of the button is displayed. In the Actions dialog, select "Play a sound". Then select "In Use" and "glissando". You may click on the Play button to hear the sound file. Make sure "sound is stored in the stack" is selected. Click OK and Done. Click on your button (using the browse tool) to hear the glissando.

Now edit the button so that it also plays an animation. Use the arrow tool and double-click on the button. In the Actions dialog, select Play Animation. Select Disk File and navigate to the TI-MEART folder and TROMBONE.PIC. Use the lasso to select the image of the trombone slide.
Next, move the mouse to place the slide "on" the trombone in the 1st position. Click the mouse button and drag the slide to the right of the screen to create the animation's path. Press any key on the keyboard to stop recording the path. Make sure Show First Frame is checked and that Erase When Done is not checked so that the slide will not disappear from the screen. Click OK and Done. Try your button.

Add one more button to the card titled Play Movie. In the Actions screen, select Play a Movie or Video, then Disk File. Navigate to the TI-MEART folder and select MARYTBN.MOV and click OK.

Save your work by selecting the "Save Stack As..." option from the File menu. Navigate to your storage medium and save your stack (end name with .STK)

Quit HyperStudio.

Stack-Based Interactive Multimedia Authoring
In-class Projects
STACK 3

Instructions - Card 1
In this project, you will add actions to text objects, graphic objects, cards, and the stack. Actions in this project include playing a CD excerpt, a MIDI file, and visual transitions.

Type your name and the date in the field at the bottom of the screen. Also type a title for the stack in that field. You may use the different text commands under the Options menu to change the appearance of the text.

After typing in your text, click on the text object with the arrow tool and choose Edit this Text Object from the Edit menu. Check the box beside Read Only to lock the text that you typed and prevent users from changing your text.

Click the Next Card button to continue.

Instructions - Card 2
First let's add an action to a button. Insert an audio CD into the CD-ROM. Use the arrow tool and click on the CD-ROM graphic below to select it and choose Edit this Graphic Object from the Edit menu. Click Features and click Make Draggable. (You must have selected "I'm an experienced HyperStudio user" in the Preferences option under the Edit menu for this to work correctly.)

Next, create an invisible button to fit over the CD player graphic. In the Button Appearance screen, click on Features and check Drop Off Only to make the button action occur when an object is dragged upon it. Click OK and fit the button over the CD player. Click on the card.

In the Actions dialog, check New Button Actions, then select the CDPlay NBA and click Use this NBA. A CD control panel will appear which is used to set up the length and volume of your
excerpt. Play the CD using the transport controls on the panel. Click "Set Start" and "Set End" to set
the beginning and ending of the excerpt. Click "Try It" to see if it is correct. If so, click on "Keep",
OK, and Done. Now when you use the browser tool to drag the CD into the player you should hear
music.
   Now you need to create a text object on this card instructing the user to drag the CD into the CD
player to hear the music. Also type in some information about the recording and to remind the user that
the CD must be in the computer. Remember to make the text object read-only when you are done.

   Click on the Next Card button to continue.

Instructions - Card 3
   Now add a text object telling the user to click
on the computer image to hear a MIDI file.
   Next, use the arrow tool and select the
computer graphic to edit it. Click on Actions
and select Play a movie or video. Then
select Disk File (QuickTime Movies) and
navigate to the TIMESNDS folder and select
the standard MIDI file, JOY.MID.
   Click on Convert and save the file as
JOY.MOV in the folder where you are keeping
your project stacks. Click on OK, OK, and
Done. The MIDI file should play when you
click on the the image.
   Now, make sounds play when the user goes
to this card. Select About this Card from
the Objects menu, then click on Things to Do When Arriving at this Card and set the Actions
to Play a sound.
   Make the stack play a sound when it is opened. Use the About this Stack option under the
Objects menu. You must open the stack after saving it to test the action.
   Now, add some visual transitions. Edit the Next Card and Previous Card buttons. In the
Actions dialog, click twice on the darkened circle beside either Next Card or Previous Card and
select a transition from the Transitions screen.
   Save your work!

Stack-Based Interactive Multimedia Authoring
In-class Projects
STACK 4

Instructions - Card 1
   This card contains a background graphic to
guide you where in placing text objects on the
screen. Every text object you create should be
a grouped object so that it will appear on every
grouped card you will create.
   For each text object graphic that has red
text, place a text object upon it and click on the
screen. Uncheck Draw Scroll Bar and
Scrollable. You may also wish to hide the
frame. Click Features and select Group
Object. Type text into each of those text
objects. Then edit the objects to make them
Read Only.
   For each text object with gray text, place a
text object upon it then click on the screen.
Uncheck **Draw Scroll Bar** and **Scrollable** for all of them except the bottom object. Do show frames. Click **Features** and select **Group Object**.

Create a button to find text and place it over the graphic below the Notes read-only text object. Make sure that its Features include **Group Object**. Choose **New Button Action** for its Action. Select the MenuChooser NBA and type *Move,Find Text* into the box and click **OK**.

Now clean up the appearance of your card as needed by erasing any portions of the background which are showing. Use either the eraser tool or the **Erase Background** command.

Add additional cards by going to the **Edit** menu and selecting **Ready Made Cards, Group Card**. (Or use the Add Record Button which does the same thing through the MenuChooser NBA.) You may then enter your information on each card!

On one of the cards, add a hypertext link that moves to another card with related information. Use the browse tool to highlight the linked text. Choose **Hypertext Links** from the Objects menu. Click **Add Link**, check **This Field Only**, click **Action**, and give the link an action that goes to **Another card** or uses the NetPage NBA to go to an Internet site (requires a web browser).

Add a title card to your stack. (**New Card** or **Ready Made Card** under **Edit**.) Under the **Extras** menu choose **Title Card** to make that card the first card in the stack.

Delete this field (the one you are reading) and the Question Mark button by selecting the object with the arrow tool and pressing the delete key. Save your work!

---

**Stack-Based Interactive Multimedia Authoring**

**In-class Projects**

STACK 5

---

**Instructions - Card 1**

In this project you will create a linear listening guide for a musical composition—sort of like MTV! It will play an audio CD. The user may choose to hear a specific excerpt or hear the entire composition.

Replace the text on this card with the proper information. Edit the objects to appear as you wish. The Next button goes to the Next card, so does the Play button, the Quit button uses the MenuChooser to Quit HyperStudio.

Divide the composition into a number of sequential sections and decide what you wish to display on the screen during each section. Create a number of grouped cards corresponding to the number of sections.

On each sequential card, place graphics, text, and/or animations to be displayed during that portion of the composition. Do not change the background of a card unless you want to change the background on every card!

On one of the grouped cards, create the following buttons. Make sure that they are grouped so they will appear on each of the grouped cards: a *Next Card* button that goes to the next card, a *Stop Play* button which uses the CDPlay NBA to stop the CD player, a *Previous* button which goes to the previous card and stops the CD, and a button which takes the user back to the title card.

Create a *Play* button which will play an excerpt from the CD. Copy and paste the button on each card. (Don't group it, since each card's button needs a different starting time.) Then go to each card and edit the button to start the CD at the beginning of that section and stop it at the end of the composition. Be sure to write down the starting time of each section.

Go back to the title card and edit the *Play* button on the title card so that it plays the composition from the CD and goes to the next card.

Now set up the timing for each card. Create an invisible button on each card which has an
Automatic timer action that goes to the next card after the number of seconds in that section of the music. (Use "Do these actions after card is shown activate after __ seconds" and "Next card".)

Go to the title card and click Play to see your linear slide show.
In this project you will create a branching stack to teach musical concepts based upon a musical composition.

Replace the text on this card with the proper information. Edit the objects to appear as you wish.

Click on the Menu button.

Instructions - Card 2

On this card, edit each button's name to create a menu of choices relating to the composition. Each button should list a separate concept. Next, create 4 cards, each of which should present one of the concepts. Use all your skills to present information through text, graphics, animation, and audio. Each card should have a shared background and a button to take the user back to the menu card.

Back at the menu card, add actions to each of the concept buttons to take the user to the related card. Also edit the Listening Guide button to take the user to PROJECT5.STK. Remember to save your work! (It would also be a good idea to add a button to the first card of Project 5 which takes the user to this stack.)
Instructions - Card 1

In this project you will create a quiz based upon the concepts presented in Project 6.

Edit the Listening Guide button to go to PROJECT5.STK. Edit the Tutorial button to go to PROJECT6.STK (Projects 6 and 7 should also have links to each other and this stack.)

Edit this card to appear as you wish. Edit the Take Quiz button to have a Testing Functions action which Asks for the user's name and give it an action that goes to the Next Card.

Go to the next card.

Instructions - Card 2

Create a number of cards corresponding to the number of topics in the tutorial stack (PROJECT5.STK). Each of card should present a quiz question through the use of a text field and/or recorded speech (or other means such as video). The card should also instruct the user to how to select an answer.

The answer choices should be presented as buttons, read-only text objects, or graphic objects. Each answer choice object should have it’s own name and a Testing Functions action as being either a correct or an incorrect answer.

Each answer object should also give reinforcement to the user such as a sound effect and showing a graphic. Each choice should also take the user to the next card.

The results of the quiz will appear in the file "hstest.txt" which will be created in the folder of this stack.

Click here to go to an example of a quiz stack.
Stack-Based Interactive Multimedia Authoring
In-class Projects
STACK 8

The following screens show a sample quiz.
Stack-Based Interactive Multimedia Authoring
In-class Projects
QUIZBUTS STACK

This stack contains buttons which demonstrate various testing functions.

Stack-Based Interactive Multimedia Authoring
In-class Projects
KARAOKE STACK

This stack demonstrates a technique for highlighting music as it plays.
Instructions - Card 1
Playing Sound Files Embedded in a Stack
First open the script editor by selecting About this Stack... from the Objects menu. Arriving at this stack... Use HyperLogo... Type the following procedure into the editor and close the window to save the script into the stack.

```
To PlaySound :snd :vol
    CBSETCMD 55
    CBSETPARAMINT 1 0
    CBSETPARAMINT 2 :vol
    CBSETPARAMSTR 3 :snd
    CALLBACK
END
```

Next, create four buttons that play sounds. Each button should have a name. Then use the Features option to hide each button. Save this stack and reopen it so the PlaySound procedure will work.

Now the stack has a procedure which can be called by any object and it has four sounds embedded. Test the procedure by creating a new visible button with a HyperLogo PlaySound procedure call: PlaySound "Soundname Volume# (e.g. PlaySound "Flute 255). Click the button to test it.

Now try playing each sound in order with a WAIT command in between (see following example).

```
PlaySound "Flute 255
WAIT 30
PlaySound "Trumpet 200
WAIT 30
PlaySound "Horn 255
WAIT 30
PlaySound "Drum 230
```

Go to the next card for more fun!

Instructions - Card 2
Using a List to Save and Play a Sequence of Sounds
First open the script editor by selecting About this Stack... from the Objects menu. Arriving at this stack... Use HyperLogo... Type the following procedure into the editor after the PlaySound procedure and close the window to save the script into the stack.

```
MAKE "soundList []
```

Reopen the stack so the script will work.

Next, create four visible buttons. Type a HyperLogo script into each of them to play one of the four sounds and add the sound to the sound list:

```
PlaySound "Soundname Volume#
```
Make "soundList LPUT "Soundname :soundList

Now create a button to play back the list of sounds. Type in this script:
IF EMPTYP :soundList [TOPLEVEL]
PlaySound FIRST :soundList 200
Wait 30
WHILE NOT EMPTYP :soundList [MAKE "soundList BUTFIRST :soundList
IF NOT EMPTYP :soundList [PlaySound FIRST :soundList 200 WAIT
30]]
(See Chapter 6 of the Exploring HyperLogo Tutorial by Bill Lynn for more information.) Test your buttons!
PlaySound "Flute 255
WAIT 30
PlaySound "Trumpet 200
WAIT 30
PlaySound "Horn 255
WAIT 30
PlaySound "Drum 230

Stack-Based Interactive Multimedia Authoring
In-class Projects
CEL STACK

A Cel Animation Metronome
Use the Export Screen command under the File menu and name the file CEL.01.
Export the screen again, naming the file CEL.03. Use the line tool to create another line whose base is at the same location as the original line, except that the line slants to the right. Erase the original line using the eraser tool. Export the screen as CEL.02. Paint another line starting from the base that slants to the left. Erase the original line. Export the screen as CEL.04.

Use the arrow tool to select and edit the My Cel Metro button. Click on Actions and check Play an animation. Click on Disk File and navigate to CEL.01. Use the rectangular selection tool and select a rectangular area around the black line that is large enough to display the slanted lines you painted.

In the Animation dialog check Show First Frame, set the Speed to a low number such as 4, and check Play non-stop. Check Transparent Color then click on the white area of the preview. Click OK and Done.

Change to the browse tool and click your button to see the cel animation.
Stack-Based Interactive Multimedia Authoring
In-class Projects
TI-MECARD STACK

Instructions
(Adding Clip Art)

Under the File menu, select Add Clip Art and navigate to the "TI-MEART" folder and select "MUSIC.PIC". In the Clip Art screen, use either the Selector Tool or the Lasso to select a musical symbol then click on "OK". While the red flashing dashes are surrounding the graphic object, you may drag it around the screen until it is in the desired location. You may also resize the graphic while the dashes appear by dragging a corner or side of the selection. You may also try some of the Effects under the Edit menu.

Click outside of the graphic. Now it may be edited with the paint tools under the Tools menu, or it may be reselected with the selection tools and changed by dragging or by using effects.

Try painting your own graphics with the paint tools.

Save your work by selecting the "Save Stack As..." option from the File menu. Navigate to your storage medium and name your stack with up to eight letters from your last name followed by .STK (e.g., WALLS.STK).

Quit by selecting the "Quit HyperStudio" option from the File menu.
PART 3

Time-Line Multimedia Authoring

David Sebald

Introduction
This outline is designed for a 15 to 20 hour unit of a graduate level course on multimedia authoring for music educators. The primary development environment of the unit is Macromedia Director, but any multimedia authoring package based on a time-line metaphor could be used. The unit is intended to complement a unit focusing on a multimedia presentation package and the basics of educational product design. Together these components comprise a 2 to 3 credit elective course in a curriculum leading to TI:ME level 2 certification. This unit is intended to be an alternative to a similar unit on a stack-based authoring program as described in part 2.

Prerequisites
Enrollees in this course should have at least basic computing skills at the level of word processing and familiarity with a graphic user interface (Windows or Macintosh OS). Concurrent or previous enrollment in a course emphasizing digital media acquisition and manipulation would also be helpful.

Description
The primary focus of this unit is the development of inservice teachers' (ISTs') skills in multimedia product creation for use in music education. Because of Director's strengths in animation and interactivity, concentration will be placed on producing both non-interactive and interactive programs targeted primarily for computer assisted instruction and general artistic creativity in public school music situations. Typical applications for the skills acquired in this unit would include: 1) short instructional animations and videos, 2) media rich CAI lessons, 3) interactive tutorials, 4) simulations, 5) instructional games, 6) public relations CD–ROM's, 7) multimedia quizzes, 8) drill & practice programs, 9) creative music videos. In order to maximize productivity in the short time span allotted to this unit, the instructor should make extensive use of prepared tutorials and templates.

Objectives
Declarative Knowledge
The IST demonstrates understanding of concepts and terminology related to current multimedia production techniques for interactive and noninteractive packages
The IST demonstrates understanding of concepts and terminology related to the appropriate application of various aspects of instructional multimedia.
**Procedural Knowledge**
The IST demonstrates the ability to apply basic production techniques of instructional multimedia development for several interactive and non-interactive packages. The IST develops a short, unique multimedia instructional package for music education.

**Assessment**
Several tutorial products receive a rating and critique based on demonstration of concept mastery and individual creativity. One final unique product is rated and critiqued on professionalism, suitability, and creativity.

**Time-Line Multimedia Authoring**

**COURSE TOPICS**

**Topic #1**

**Multimedia and Time-Line Multimedia Authoring**
The instructor demonstrates several typical multimedia products made with Director. The instructor leads the class in a discussion of multimedia considering the following concepts:

1) What is multimedia?
2) What are its appropriate educational functions?
3) How can it be effectively applied in a music classroom?
4) What hardware and software tools does its production require?
5) What are the most important guidelines for creating educational multimedia?

The instructor presents a brief overview of Director demonstrating the following concepts:

1) Its place as the leading development package for web & CD-ROM production.
2) Its time-line metaphor.
3) Its most used windows and controls.
4) Its strengths (animation, interactivity, extensibility, cross-platform, no license fee)
5) Its weaknesses (expense & learning curve).
6) Classification of Director products. (Time-Line Handout 1)

The instructor distributes sample Director source files for perusal.

**Class Activities #1**

ISTs independently view, critique and categorize numerous individual examples of Director source files from CD-ROM (Time-Line Handout 2)
Topic #2
Time-Line Multimedia Authoring Basics
The instructor distributes and demonstrates the Director Basics Tutorial (Time-Line Handout 3). This tutorial covers:

1) Identifying names and functions of the most often used components of Director 6.0. This includes icons, windows, menus, gadgets, and command key combinations.
2) Manipulating all of the above at a basic level. (move, resize, open, close, scroll, etc.)
3) Setting up a stage.
4) Creating visual cast members (bitmaps, shapes, and text blocks)
5) Placing cast members on the stage in several ways.
6) Creating basic tweened animations.
7) Playing the animation with a controlled frame rate and manual step control.
8) Saving and analyzing a Director project.

The instructor demonstrates visual and aural media acquisition and distributes guides specific to the hardware and software to be used by this class. (Time-Line Handout 4) Concepts include:

1) Clip art (visual and aural)
2) Using Director's Paint window.
3) Scanning/Digitizing and manipulating still images.
4) Capturing video.
5) Capturing audio.

Class Activities #2
ISTs work independently on Tutorial #1 and develop a unique variation of its product
ISTs digitize and save photos/graphics using scanner and or digital still camera.
ISTs digitize and save audio from line source or microphone.

Topic #3
Noninteractive Animation 1
The instructor demonstrates and classifies basic types of animation possible in Director. (Time-Line Handout 5) These include:

1) Self running slide show with transitions.
2) Real time generation.
3) Straight line (X & Y plane movement).
4) Z plane (resizing).
5) Curved paths.
6) Acceleration/deceleration.
7) Blending.
8) Color cycling.
9) Sprite Distortion.
10) Page flipping.

The instructor demonstrates various methods for incorporating audio (sound effects, narration, and/or music tracks) into Director movies. This demonstration covers the appropriate application of:

1) Various audio formats (aiff/wav/au, Shockwave audio, QuickTime audio, [MIDI], Beatnik).
2) Types and levels of compression. (bit resolution, sampling rate, SWA level, choice).
3) Synchronization with visuals ([SMPTE], [frame rate], partition, cues).

The instructor demonstrates Director's built in progression controls (Timer channel pauses & cues).

Class Activities #3
ISTs develop their own variations of animation templates. (Time-Line Handout 5)

---

**Topic #4**
**Noninteractive Animation 2**
The instructor demonstrates linear storyboarding.
The instructor demonstrates sample music videos created in Director and their storyboards.

Class Activities #4
Using provided templates, ISTs develop storyboards for a short music video or narrated story. (less than 2 minutes) (Time-Line Handout 6)
Under guidance from the instructor and using their previously created storyboards, ISTs develop original music videos or noninteractive movies which incorporate images (photos or graphics), several types of animation, and audio (music, sound effects and/or narrator).

---

**Topic #5**
**Interactive Multimedia Product Design**
The instructor demonstrates the difference between interactive and noninteractive instructional products.
The instructor demonstrates principles of effective interactive product design and distributes written guidelines. (Time-Line Handout 7) Concepts include:

1) Typical product development sequence.
2) Screen design guidelines.
3) Instructional interface effectiveness guidelines.
Class Activities #5
ISTs and instructor discuss guidelines.
ISTs sketch screen layouts that meet and break design rules.

Topic #6
Scripting and Interactivity 1
The instructor introduces Director's Lingo. (Time-Line Handout 8)
The instructor discusses and demos basic programming concepts including:
   1) linear nature of computer processes
   2) branching vs linear decisions and flowcharting
   2) events (on exitframe, mouseup, mousedown)
   2) conditions (if..then.. and its variants)
   3) loops (repeat with, repeat until)
   4) handler hierarchy (sprite, cast, frame, movie)
The instructor demonstrates several important navigation handlers including: (Time-Line Handout 9)
   1) on exitframe | go to the frame | end
   2) on mouseup | go to the frame+1 | end
   3) on exitframe | go to the frame-1 end
   4) on mouseup | go to frame # | end

Class Activities #6
ISTs develop their own variation of interactive tutorials 1 - 5 (Time-Line Handout 10 - 13a&b)

TOPIC #7
SCRIPTING AND INTERACTIVITY 2
THE INSTRUCTOR DEMONSTRATES THE CREATION AND USE OF INTERACTIVITY TOOLS
FRAME MARKERS
BUTTONS AND SLIDER CONTROLS
PUPPET SCRIPTS (FOR SOUND AND CAST MEMBER SWITCHING)
INTERACTIVE TEXT FIELDS
PROGRAMMERS FEEDBACK (THE MESSAGE WINDOW)
OTHER DEBUGGING TOOLS
VARIABLES (LOCAL & GLOBAL, INITIALIZING

Multimedia Authoring - Ti:ME 2B - Teacher Manual
Page 52
THE INSTRUCTOR DEMONSTRATES SEVERAL MORE USEFUL SPECIFIC SCRIPTS: (TIME-LINE HANDOUT 7)

ON MOUSEUP | GO TO FRAME [# OR "NAME"] | END
ON STARTMOVIE | SET VARIABLE = | END

CLASS ACTIVITIES #7
ISTS DEVELOP THEIR OWN VARIATION OF PROVIDED INTERACTIVE TEMPLATES
(TIME-LINE HANDOUT 14a)

**TOPIC #8**
**SCRIPTING AND INTERACTIVITY 3**
THE INSTRUCTOR DEMONSTRATES SCRIPTING AND APPLICATION OF MORE ADVANCED PROGRAMMING CONCEPTS
INCLUDING
  1) RANDOM FUNCTIONS
  2) USEFUL MATH FUNCTIONS
  3) SETTING UP AND ACCESSING LISTS (ARRAYS)
  4) CREATING AND USING TIMERS
THE INSTRUCTOR INTRODUCES GUIDELINES FOR THE FINAL PROJECT.

CLASS ACTIVITIES #8
ISTS DEVELOP SHORT DEMOS OF THE ABOVE CONCEPTS FROM SCRATCH OR PROVIDED TEMPLATES. (TIME-LINE HANDOUT 14b)
ISTS PLAN AND SKETCH STORYBOARDS OR FLOWCHARTS FOR FINAL PROJECTS.

**Topic #9**
**Delivery Considerations**
The instructor demonstrates and discusses various methods of delivering Director products including
  1) Source files & players.
  2) Protected files & players.
  3) Projectors.
  4) Shockwave.
The instructor demonstrates and discusses various media for delivering Director products including:
  1) Hard drive / Zip.
  2) CD-ROM.
  3) Internet.
  4) LAN.
The instructor discusses packaging considerations for Director products.

Class Activities #9
ISTs hand in refined storyboards/flowcharts.
ISTs work on final projects
**TOPIC #10**
**EXTENDING DIRECTOR**

The instructor demonstrates popular Director Xtras and complementary programs (Flash, FreeHand, Photoshop, modeling programs).

Instructor provides sources for Director Xtras.

**CLASS ACTIVITIES #10**

ISTS work on final projects.

---

**Topic #11**

**Students & Time-Line Multimedia Authoring**

The instructor will lead a discussion of curricular integration of multimedia and copyright issues. Instructor demonstrates public school students' creative projects. Instructor leads class in a discussion of the place of multimedia in public education. Instructor and class informally discuss the nature of creativity and its nurture in students.

**Class Activities #11**

ISTS work on final projects.

ISTS complete TIME-LINE HANDOUTs 15a and 15b.

---

**Topic #12**

**Final Product Demo & Critique**

Instructor hands out critique sheets and reviews critique parameters.

**Class Activities #12**

ISTS demonstrate final projects individually and collect critique sheets.

ISTS complete critique sheets on class.
Student Worksheets and Handouts - Part 3
Time-Line Interactive Multimedia Authoring
HANDOUT 1

Categories of Time-Line Multimedia Applications

Typical applications for the skills acquired in this unit would include the following. Briefly describe each.

1) short instructional animations and videos

2) media rich CAI lessons

3) interactive tutorials

4) simulations

5) instructional games

6) public relations CD–ROM's

7) multimedia quizzes

8) drill & practice programs

9) creative music videos.
Time-Line Multimedia Handout 2: List of Director Source Files Available for Study

Each instructor will probably have his or her own collection of sample Director source files collected from Macromedia Demo CD's, Director Books, personal products and former students' products. The following is a list of those made available to multimedia production students at UTSA.

AFRICA.DIR: MediaBook Sampler newBreakfast.AIF
Alternating Graphic 2 menu Jumping Demo
ANCITOR.DIR: MENU.DIR Running Demo
ANETTEX.DIR: Meteor Shower Somaunti Demo
Apple Demo: MM1.DIR buzzhead.dir
audionymous: MOONRIS1.DIR buzzhead.prj
Ball Catch: More Puppets buzznot.dir
BALLSHIR.DIR: More Special Effects DEEP2.AIF
BAILSTRE.DIR: NOISE1.DIR leary.pct
BANDX.DIR: ONLINE.BAT ask.dir
Basic Leaps: Other Dolphin talk1.dir
BRAILL1.DIR: Presentations talk2.dir
be3.dir: paddleball.dir tell.dir
behav1.dir: PHOTOSH.DIR text.dir
BELLTOUR.DIR: planets dir win.dir
BEWARE1.DIR: Popup Menus Window Test
Bird MOV: PremMac.dir GunnedD.DIR
bird1.drv: PRESENTA SJ Merc Sign East.movie
Black Jack: Progbbf SJ Merc Sign South_movie
blendx3.dir: earthTrans.808bit
BONOBUT.DIR: proximity test dir earth1.png
BRIXXX.DIR: psychelic earthmiaw.dir
Button Navigation: Puzzle earthRot.dir
buzzhead: givrMac.dir earthrot1.png
CARACCEL.DIR: Racing Sprites linkedearthrot.png
ch27: RADIOM.GAT rotQT.mov
Check&Radio: REDSIO.DIR video pred dir
CHECKIT.DIR: RESUME.DIR ELLJOTT.DIR
CHICAGO.DIR: Rolling w/Buttons done BOVERRO.MOV
CKS: ROLLOVER.DIR BOVERHO.DIR
CLOCK.DIR: SANCWOD.CIR CANTERIO.DIR
CLOCK.OPEN.DIR: Scroll Bitmap - C1H1
COLORS.DIR: Horizontal C1H2
Convert Temp done: Sebald-99 C1H3
DANCERS2.DIR: Selection Lists C1H4
DANCSN5DIR: SHARP0.DIR C1H5
DIA.mov: Simple Database C1H6
DIRECTOR.DIR: Simple Frame Script C1H7
Drawing with Trails: Simple Hypertext C1H8
earth rotation: SKETCHBK.DIR C1H9
ELLJOTTD SLIDER.DIR LESSON01.DIR
EXAMPLE1.DIR: Sliders LESSON1.DIR
EXAMPLE2.DIR: Sliding Puzzle LESSON2.DIR
EXAMPLE3.DIR: SOLAR.HOT LESSON3.DIR
FACTORY.DIR: Sonic 3D tip LESSON4.DIR
Faking 3D: SoundEdit.dir LESSON5.DIR
FILES: SSL.VID.DIR SHARP1.DIR
fileO: stars.dir Clicdoub
Flash Sprites: States Clicdoub
GAUGE3.DIR: STORY3.DIR Clicdown
GEAR65.DIR: STURN.DIR ClicUp
Globals: TABLESSET.DIR F101.DIR
GOBOOM.DIR: TECHNORM.DIR F102.DIR
GREEK4.DIR: THENGIR.DIR F103.DIR
GUIDE.DIR: timeout F104.DIR
HIBUTTON.DIR: toolbar1.dir F105.DIR
HOUSEJAM1.DIR: trails2 F106.DIR
HOUSEJAM2.DIR: UDOMAM1.DIR F107.DIR
HYVIEW.DIR: video dir F108.DIR
Imagination: VISUALZ.DIR F109.DIR
Ink Effects Demo: WALKMAN2.DIR player
intranm.dir: WOODSTOK Clicdoub
IASKH3MOCZAVECZ: IANs.WWDC
JOBSEEK: IAN's WWDC Clicdoub
KEY3D: IAN's WWDC proj ClicUp
KIOSKI.DIR: LegalSysSio Traffic EXAMPLE1.DIR
KIOSK1.SWA: Movie EXAMPLE2.DIR
MAC: LegalSysSio v2 EXAMPLE3.DIR
MARI00L7.DIR: maps F201.DIR
Matching: audionymous F202.DIR
MATCHET.DIR: audionymous demo F203.DIR
MCMIDH.DIR: moreinfo.QT F204.DIR

Multimedia Authoring in Director (Ti:ME 2B) Page 57
Objectives:

**By the end of this tutorial you should be able to:**
1. Identify names and functions of the most often used components of Director 6.0. This includes icons, windows, menus, gadgets, and command key combinations.
2. Manipulate all of the above at a basic level. (move, resize, open, close, scroll, etc.)
3. Set up a stage.
4. Create cast members
5. Place cast members on the stage in several ways.
6. Create basic tweened animations.
7. Play the animation with a controlled frame rate.
8. Save and analyze a Director project.

Tutorial:

1. **Set up the environment and open Director.**

Most computers can have several programs loaded into RAM at the same time. Director works best when no other applications are running, especially on computers with limited memory. Check for this now and, if necessary, clear any loaded programs from memory. Here's how to do it:

   a. Move the pointer to the farthest right icon on the menu bar. This is the icon for the **Application Menu** and will usually look like a classic Macintosh computer.
   b. Hold the mouse button down on the icon.
   c. If you see any words other than "Finder," drag the pointer down to that other name (like "MusicShop" or maybe "Finale") to highlight it and then release the mouse button. This action activates that program. Then hold the command key down and press the "q" key (**Cmd - q**) to quit the program.
   d. Repeat this process for any programs other than the Finder that might be taking up memory.

When "Finder" is the only word left showing under the Application Menu, you are ready to open Director. Do this now like this:

   a. Move the pointer to the **Apple Menu** in the upper left corner of the screen.
   b. Hold the mouse button down and pull down to **Applications**.
   c. Still holding the mouse button down, pull over and down to **Director 6.0**.
   d. When Director 6.0 is highlighted, release the mouse button. The program will open.
2. Explore Director's major windows and controls.

The five most often used parts of Director's visual interface are: 1) **stage**, 2) **control panel**, 3) **cast window**, 4) **score**, and 5) **paint window**. All of these parts can be brought up through the **Windows** menu, but it's a lot more efficient to learn the Command key combinations. That's what we'll do. Manipulate these now one at a time to get a feel for how they work and read below what their basic functions are:

a. **Cmd - 1** clears the **stage** of the menu strip and any open windows. Another press brings everything back into view. The stage is where visual things happen; images of cast members called sprites are placed here and moved or metamorphed through time. No matter how many windows are open, this command makes all those windows invisible so that entire stage can be seen.

b. **Cmd - 2** opens or closes the **control panel**. The control panel looks and acts like a video cassette's controls with obvious play, stop, rewind buttons.

c. **Cmd - 3** opens or closes the **cast window**. All individual elements of a multimedia presentation--graphics, sounds, movies, animation loops, buttons, text, palettes, command scripts, etc.-- are placed in separate boxes of the cast window.

d. **Cmd - 4** opens or closes the **score**. The score is where the cast members are combined and caused to move or change according to the producer's directions. Horizontal rows called **channels** represent separate cast members appearing at various times throughout the length of a presentation. Vertical columns called **frames** represent time units that can be set to anything from 1/120th of a second to indefinite periods controlled by the user. Each individual rectangle is called a cell. Director 6.0 allows 120 sprite channels plus 2 channels for sounds and one each for transitions, command scripts, palette changes, and timing (called **tempo**). The number of frames is essentially unlimited; it keeps growing as currently available frames are used.

e. **Cmd - 5** opens or closes the **paint window**. This is a surprisingly powerful and easy to use image creator. Images painted here are automatically inserted into the next available box in the cast window; however, unlike some paint programs, only a area containing the actual painted part is inserted into the cast window's cell.
As you might notice if you hit any of the other command-number key combinations, there are several other windows--text, tools, palettes, video, script, and message; however these are normally less used. Later demos and individual help with your project will demonstrate their functions.

3. Set the stage.

Move the pointer up to the Modify menu. Pull down to Movie and over to Properties, and release the mouse button.

In the dialog box that comes up, hold down on the color square as in the picture shown at right. Select a dark shade of blue and release the mouse button. The stage turns blue.

4. Create the cast members.

Press Cmd - 5 to bring up the paint window. Click and hold on the foreground color square. Select brown. Choose "Normal" from the ink pop-up menu. (see 1)

Select the dotted top line (no line) from the line width selector. (2) Select the rectangle tool. (3) Pull out a slim rectangle across the width of the paint window.

Press Cmd - 3 to see it as a thumbnail in the cast window. Then press Cmd - 3 again to close the cast window.

Click on the plus sign at the top left of the paint window to select a new cast box. (4) Select the ellipse tool. (5) Choose "Gradient" from the ink pop-up menu. (6) Select red for the left color gradient square. (7) Select yellow from the right gradient square. (8) Double click the ellipse tool to open the gradient dialog box. Choose "Sun Burst" direction. Choose "One" cycle. Choose "Dither Best Colors" method. Choose "Equal" spread. Choose "Paint Object" range. Click on the OK button to close the dialog box and return to the paint window.

Hold down the shift key (to constrain the ellipse to a circle) and pull out a ball about 1 1/2" in diameter.

Press Cmd - 3 to see the ball in the cast window. Press Cmd - 5 to close the paint window.
5. Place sprites of the cast members on the stage.

Open the cast window (Cmd - 3) if it is not already open. Drag it to the upper part of the stage so that it will be out of the way for the next operations.

Click and hold on the box with the rectangle. Drag the outline ghost of the image down to the lower center of the stage. Notice that the ghost assumes the proper size as soon as it is dragged over the stage. Release it there. Likewise, drag the ball to the upper left of the stage.

Press Cmd - 3 to close the cast window.

(Check procedure 10 at the end of this tutorial if you want to get photographs, QuickTime and sounds into Director's cast.)

6. Create motion frames.

Press Cmd - 4 to open the score. It should look like the image at right. Notice that cast member one (the rectangle) is in sprite channel 1. That's because it was the first one you moved onto the stage. You should have about 30 frames darkened for each sprite. (It's easy to lengthen or shorten it by dragging the end.)

Click and release on the last frame darkened in sprite channel 2 as you see here. The first frame and last frame of the dark area are called key frames. You're about to move the ball to a new position on this key frame.

For this next step, you may have to press Cmd - 1 to make the score window invisible in order to see the whole stage.

Now, while the last key frame is selected, go back to the stage and drag the red dot at the center of the ball down so that the ball touches the rectangle. You've just "tweened" all the frames between your first (green) and last (red) key frames

To test the animation so far, press Cmd-option-r to rewind the movie to the beginning. Then press the enter key (not the return key) to play all the frames. If you don't have an enter key on your keyboard, use Cmd-option-r. Press Cmd - . (the period key) to stop the animation if it continues to loop. To toggle looping on and off, click the loop button on the control panel.

7. Finish the animation.

Open the score window (Cmd - 4). You probably had a white rectangle surrounding the ball. Let's remove it before going on. Click the middle of the channel 2 sprite and release to select the entire sprite. Then move the pointer to the Ink button at the top of the score window. Hold down on the button and release on "Matte." Voila! the rectangle is now gone.
We're going to make copies of the two sprites now. 
Holding the option key down, click and hold on the channel 1 sprite. Drag the copy to the same channel just after the first sprite. (If you didn't hold the option key down, you would simply move the sprite rather than moving a copy.)

Do the same thing with the channel 2 sprite. When it's in place, leave it selected and go back to the stage. (Press \text{Cmd} - 1 to make the score invisible.) Drag the red dot (the last key frame) of the tweening line to the upper right of the stage. Drag the green dot (the first key frame) down to the place where the previous sprite's last key frame was. You should now have a mirror image of the first sprite's motion.

To see the entire animation, rewind the movie (\text{Cmd}-\text{option}-r) and run it (\text{enter}).

8. Save the project.

Insert your disk into the floppy or Zip drive. From the File menu choose "\text{Save As}.
Click on the \textbf{Desktop} button. (1) Double click on your disk's name to open it. (2) Click in the title box and type in a title. (3) Click on the \textbf{Save} button. (4)

9. Clean up.

Press \text{Cmd} - q to exit Director. Drag your disk icon to the trash to eject the disk.

(10. Get photographs, QuickTime movies and sounds into Director.)

- If you want to use externally produced media like scanned images or digitized sounds in your Director movie, press \text{Cmd}-r to open the Import dialog box
- Navigate until you find the file you want in the top window.
- Double-click it to move it to the lower window. You can select several files at a time this way.
- When you have all the files you want at this time, click on the \textbf{Import} button. They will be loaded into the cast window and become part of your movie.
- Be a little careful of the amount of memory you consume doing this. Photographs and sound files are often huge and may slow your animations down.
- Because Director doesn't directly play MIDI files, it's better to convert them to QuickTime movies which will use the Macintosh's sound resources and be about a hundred times smaller than sound files.
Time-Line Interactive Multimedia Authoring  
HANDOUT 4  
Media Acquisition Guides

These guides will be specific to each instruction site but probably should include:

Grabbing and modifying Clip Art

Scanning and manipulating photographs

Digitizing and manipulating audio

Digitizing video clips

NOTE: The TI:ME 2B course, Digital Media, contains such guides
Time-Line Interactive Multimedia Authoring
HANDOUT 5
Some Basic Animation Templates

These templates are available online as both shocked files and source code

**StraightLineAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/straightline.dcr(2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/straightline.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/straightline.zip

**CurvedPathAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/curve.dcr(2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/curve.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/curve.zip

**Accel-DecelAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/accel.dcr (2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/accel.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/accel.zip

**Z-PlaneAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/zplane.dcr (2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/zplane.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/zplane.zip

**DistortAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/distort.dcr (3k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/distort.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/distort.zip

**BlendAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/blend.dcr (2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/blend.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/blend.zip

**PageFlippingAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/pageflip.dcr (6k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/pageflip.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/pageflip.zip

**RealTimeAnim**
Shockwave http://music.utsa.edu/multimedia/materials/examples/realtime.dcr (2k)
Source Macintosh http://music.utsa.edu/multimedia/materials/examples/realtime.dir.sit
Source PC http://music.utsa.edu/multimedia/materials/examples/realtime.zip
Describe the process of designing/storyboarding/flow-charting hypermedia productions.
These guidelines have been gathered over a period of years from many resources on instructional design. Some concepts are expressed several times in different words, and some may appear to conflict. Still, each guideline has a logical justification in some context and when applied appropriately will result in a more effective multimedia project.

**Rules for Efficient Production**

**Analyze the need:**
- What is the purpose?
- Who is the intended user?
- What is the intended platform?

**Research the topic thoroughly:**
- What is the course content?
- What are the major topics/concepts?

**Establish goals & objectives:**
- What specific concepts, skills, attitudes should the user acquire through the instruction?
- How can these outcomes be measured?

**Sequence instructional tasks:**
- Write a flowchart on paper.

**Gather all media.**

**Decide on appropriate production tools.**

**Create the computer program.**

**Package the product**
- Produce accompanying documents.
- Create covers.

**Implement the product.**

**Evaluate & revise the product.**

**Visual Interface Rules**

- **Use Contrast** to make things easy to distinguish and keep separated.
  - Allow "frame" space around the screen elements.
  - Distinguish different kinds of information visually.
  - Leave lots of room around graphics.

- **Use Repetition** to keep continuity both on a page and throughout a project.
  - Reproduce the screen exactly in each frame.
  - Keep continuity of screen elements (identity/size/placement) from frame to frame.
  - Keep element locations constant.

- **Use Alignment** to make screens sections aesthetically pleasing and to lead the viewer's eye appropriately.
  - Titles should be at top and to the left.
  - Start frame in the upper left.

- **Use Proximity** to group related items together.
  - Controls and options should be at bottom.

- **Use Balance** to keep the screen seem economical and aesthetically pleasing.
  - Center the display.
  - Use rule of thirds to place objects of importance.
  - Weight the screen equally.

- **Use Emphasis** to bring the most important things to the viewers attention.
  - Omit extraneous detail.
  - Keep readings short and logically segmented.
  - Use a 24 line screen.
  - Indicate topic and concept clearly on every frame.
  - Keep relationships to all information clear.

- **Use Logic** to put things where they make the most sense.
  - Form should follow function.

- **Use Color** for psychological reinforcement.
  - Don't use blue on black.
  - Don't use too many colors.
  - 4 - 6 colors is best for an interface.
  - Blue is always a safe background color.

- **Use Dynamics** to imply feeling and for psychological reinforcement.
  - Consider cultural values in symbol and color choice.

- **Keep the message more important than the medium.**

- **Keep the interface clean (don't let effects get in the way of clarity.)**
  - Be economical in the use of screen gadgets and controls.
  - Make sure that the viewer grasps meaning fast and easily.
  - Don't clutter visuals with too much information.
Make symbols obvious, attractive, simple, informative, distinct, concrete, familiar, legible, and few in number.
Use "normal" photographs to illustrate concept, not photos that call attention away from it.
Use animation for dynamic concepts.
Keep screens orderly.
Use easily identifiable navigation tools and icons.

The Four Rules of Screen Design

Keep it consistent.
Keep it uncluttered.
Highlight important ideas.
Keep it readable.

Keep screen consistent:
Type size and font
Element location
Color
Control location and identity
Design and mood
Use lots of blank spaces

Rules for text:
Keep text large enough to read easily.
Don't stack text.
Use serifed fonts for text to be read.
Use sans-serifed fonts for headlines.
Avoid stylized fonts except as symbols.
Limit the sizes, styles, and colors to 3.
Don't use too much text to be read quickly in any text block.
Keep text as short as possible.
Use left justification in text blocks rather than full justification.
Keep text readable.
Grade level.
Wording.
Content.
Context.
Sequence.
Style.

Highlight important concepts:
Use plain English.
Use lots of blank spaces.
Don't duplicate aural narration with visual text.

Rules for user interaction:
Let the user control action as much as possible.
Motivate the user.
Allow the user to control progression and navigation.
Allow for self-pacing but not procrastination.
All frames should contain an interactive component.
Keep the user oriented.
Frames can be multipurpose (present information & encourage manipulation).

General rules for effective instruction:

B.F. Skinner's 4 Basic Principles of instruction:
Formulate clear objectives.
Use substeps in large units.
Allow for individual progress.
Build new concepts on previously learned material.

Other Ideas:
Engage the user's attention. (enhances memory).
Present a question before presenting information. (doubles knowledge retention).
Clearly connect new information to previously acquired knowledge.
Keep information flowing (to prevent boredom).
Provide mnemonics and iconic or pictorial aids.
Ask the user to reconstruct steps leading to newly acquired concepts.
Use varied repetition & practice to cement important concepts.
Reward correct responses.
Do not use clever, sarcastic, or harsh punishments for wrong answers.
Keep the environment and problems as realistic as possible.
Keep the structure open and understandable.
Keep objectives clear and meaningful.
Keep instruction clearly sequenced, progressive, and in easily digested steps.
Avoid leaps of logic.
Don't bore the user.
Learning should be fun: Stimulating, motivating, captivating, engaging, rewarding.
Aim to the level of the user: Age, socio-economic group, familiarity with topic.
Rules for Physical Environment
Remember that reading text on paper is more efficient than reading text on screen.
Keep screen brighter than ambient light.
Keep screen contrast high.
Eliminate flicker.
Keep a center of acuity.
Eliminate head/neck strain.
Design for a 16” eye to screen distance.
Lingo is a rich and constantly growing scripting language underlying Director. With it you can control interactivity between your product and the end user, and you can create far more complex actions. Although there are literally hundreds of commands, functions, properties, operators, constants, and keywords in Lingo, you can do all the most common actions after learning to manipulate only a few of them. That is what the following short tutorials will teach you.

Some helpful concepts

Here are two basic concepts you'll need to know in order to start using Lingo effectively right away:

1. There are two main parts of every Lingo script no matter how short: 1) a test for an event happening or a condition being met and 2) a command or set of commands to execute when the test result is positive. Two common examples:

   ```lingo
   on exitFrame
     go to the frame
   end
   on mouseDown
     go to the frame + 1
   end
   ```

   Notice that every event test has to have an equivalent "end" statement to indicate where the test and subsequent commands conclude.

2. Lingo scripts must be attached to one of four movie levels in order to work: 1) a sprite, 2) a cast member, 3) a frame, or 4) the entire movie. This list also indicates the priority of execution for commands. For example, if your movie is repeating a frame over and over with the "go to the frame" command and during that loop you click on a sprite that has the "go to the frame + 1" command attached to it, the sprite command takes precedence and your movie jumps out of the loop and on to the next frame. The tutorials that follow will show you how to attach scripts to sprites, frames, and the movie. (We don't use cast member commands much in this class.)

   Here are a few other useful hints:

3. Capitalization in your lingo scripts doesn't matter.
4. Exact wording and spelling does matter.
5. It's very common for commands not to work the first time you script them. There are a lot of ways to create scripting errors.
6. It's helpful to include lots of comments to explain what commands are actually doing. Do this by typing "--" and then your comment. Anything on a line after the "--" will not be interpreted as part of a command by Director.

7. Writing simple Lingo commands is actually easier and more fun than relying on Director's included "Behaviors."

8. You can always get to the Lingo scripting window with "Cmd-0" (Command key + zero). There you can use the left and right arrow buttons to move to the command you want to work on.
Here are some of the most often used Lingo scripts. Many Director movies will require nothing more than these short examples placed strategically to achieve a fairly high level of interactivity. They can be combined in many ways.

**on exitFrame**

- **go to the frame**

  The most basic of scripts, this can be attached to any frame in the frame script channel to stop the forward progression of the play head. It actually commands the play head to continually replay the current frame.

**on exitFrame**

- **go to the frame - 10**

  Similar to the above script, this one allows for some animation by moving the playhead back a certain number of frames and letting it replay that sequence over and over.

**on mouseUp**

- **go to the frame + 1**

  Another very basic script. It is usually attached to a sprite in which case it takes precedence over the "go to the frame" scripts above and continues forward motion.

**on mouseUp**

- **go to frame 100**

  This script is for non-linear navigation. Attached to a sprite it can take the user anywhere in the movie.

**on mouseDown**

- **puppetsound 1, "mouseclick"**

  A script which plays an audio cast member. The cast member can be referred to either by name or by its member number.

**on mouseDown**

- **set the member of sprite 10 to 20**

  Attached to a sprite, this script is often used to replace an image of an "up" button with that of a "down" button.

**on mouseUp**

- **set the member of sprite 10 to 19**

  This is the converse of the scripts above. Notice that more than one command can be incorporated into the same script

**on mouseDown**

- **puppetsound 1, "mouseclick"**

  Scripts like this are often used during development to test where things are on the stage. When the mouse is clicked, the coordinates of the cursor's tip are written to the message window.

**on mouseDown**

- **put the mouseh, the mousev**

  (Commanding "put" without specifying where automatically places the output into the message window.)
This tutorial uses 2 basic Lingo commands to control the forward progression of a Director movie. Think of it as a simple slideshow or PowerPoint presentation. When you run the finished movie it will pause on each image. When you click on the image, it will move forward to the next image.

Here's the setup:

1. Create a stage sized to 320 by 240 pixels:
   a. Choose Modify/Movie?Properties from the menu strip.
   b. Choose QuickTime 320 x 240 as the Stage Size.
   c. Choose an appropriate Stage Color
   d. O.K. the Movie Properties window.

2. Create 6 different cast members using the Paint Window:
   a. Choose Window/Paint from the menu strip.
   b. Use the paint tools to create any simple image that will fit on your 320 x 240 stage.
   c. Press the "+" button at the top left corner to create a new empty space for painting the next image.
   d. Repeat b and c until you have enough cast members.
   e. Alternatively you could import some photos or clip art if you wanted, but it's not necessary to learn the techniques we're presenting here.

3. Put sprites of the cast members on stage one at a time:
   a. Drag cast member 1 to the stage.
   b. In the Score window, shorten the sprite length to 5 frames.
   c. Click on frame 6 in the same channel of the Score window.
   d. Repeat a, b, and c until sprites of all cast members are on stage in sequential order. Your Score window should look something like this picture.
Now here's the scripting part:

4. Double-click on frame 5 in the Frame Script channel. This brings up the Scripting Window with the event part of your script ("on exitFrame ... end") already written for you. All you have to do is type "go to the frame" in the blank line. Director understands "the frame" to mean the frame that the playback head is currently on. Close the Scripting Window to attach the script to frame 5. Now click once on frame 5 in the Frame Script channel to be sure the Lingo command you just created is selected. Press Cmd-c to copy it. Click once on frame 10 of the Frame Script channel. Press Cmd-v to place a copy of the command there. Do the same cmd-v on frames 15, 20, 25, and 30.

5. Click once in the middle of the first sprite in channel 1 to select it. Click and hold on the Script Popup Bar. Choose New Script and release on it. This brings up the Scripting Window with the event part of your script ("on mouseUp ... end") already written for you. Type "go to the frame + 1" in the blank line. Close the Scripting Window to attach the script to the first sprite. Click once in the middle of the second sprite in channel 1 to select it. Choose the script you just created from the Script Popup Window to attach a copy to the second sprite. Do the same with the remaining sprites. That's all; you're done. Now save your program and try running it.
This movie builds on the first interactivity tutorial, Time-Line Handout 10: Linear Control, to create a set of buttons that will take the user from any part of the movie to any other part. This is a very useful technique for giving the user more complete navigational control of the learning environment. If you didn't save the first tutorial's movie, you'll need to go through at least the first 4 steps of that tutorial to prepare a simple sequence of images. If you did save it, load it now and be sure that the play head is on the first frame.

1. Create a button for the first image:
   a. Choose Window/Tool Palette from the menu strip.
   b. Click on the Button tool in the Tool Palette.
   c. Draw a small button on the lower left side of the Stage.
   d. Type "1" to title the button and then click somewhere else on the stage to de-activate it. Your stage should look something like the one shown here when you've finished.

2. Make 5 more copies of the button cast member:
   a. Choose Window/Cast from the menu strip.
   b. Click once on the button cast member you just made.
   c. Press Cmd-c to copy it.
   d. Click once on the next empty frame.
   e. Press Cmd-v to place a copy of the button in this space.
   f. Repeat steps d and e until you have a total of 6 button cast members.
   g. Double-click each of these cast members and, in the Text Window that comes up, give each a consecutive number so that they are numbered 1 through 6. If you want to be fancy, use the background color chip to change the button's color while the text is selected in the Text Window.
   h. Close the Text Window.

3. Drag each button from the Cast Window to the Stage so that they line up as shown at right.
   Alignment can be accomplished in several ways:
   1) by simply using the arrow keys to nudge them 1 pixel at a time
      (Shift-arrow moves them by 10 pixels at a time)
   2) by clicking on a button and typing cmd-shift-i to open the Sprite Properties Window then entering appropriate numbers. (I used horizontal increments of 50 pixels after the first button.)
3) by choosing View/Grids and Show and Snap to and settings to make a visible grid on the stage to help align them. Whichever method you choose, the stage should end up looking like this.

4. Open the Score Window and drag the end of each button sprite's bar so that it is the same length as the combined image bars.

5. Select the first button's sprite in the Score Window.
   Click and hold on the Script Popup Bar.
   Select New Script and release on it. In the window that comes up type "go to frame 1" in the empty line.
   Close the window.
   Do the same for each button's sprite but change the number to that of the frame where each image starts. (The numbers on this example are 1, 6, 11, 16, 21, 26).

That's it; you're done. Now save your program and try it out. If you built this on top of the first tutorial, you should have two types of interactivity: sequential progression by clicking on the images, and non-linear movement by clicking on the buttons.
The next few tutorials will show you how to create several useful interface gadgets and also show you how to control visual changes in a manner different from previous tutorials. This movie has only 1 frame: all changes occur while it loops. There are 3 tutorials in this set which should make it easier to understand each function. This first one shows you how to make your own button rather than using the rather mundane button tool included in Director's Tool Palette.

1. Create a stage sized to 320 by 240 pixels:
   a. Choose Modify/Movie?Properties from the menu strip.
   b. Choose QuickTime 320 x 240 as the Stage Size.
   c. Choose an appropriate Stage Color
   d. O.K. the Movie Properties window.

2. Using the Paint window, create two simple buttons:
   a. Choose black from the foreground color chip.
   b. Choose a thin line size
   c. Choose the unfilled circle tool.
   d. Hold the shift key down (to constrain to a circle rather than an oval) and then draw a small circle.
   e. Choose the fill tool.
   f. Choose a red from the foreground color chip.
   g. Pour the red color into the circle. This is your first button.
   h. Now press Cmd-a to select the entire button.
   i. Press cmd-c to copy this selection to clipboard memory.
   j. Click on the + button to create a new empty work space.
   k. Press cmd-v to paste a copy of your button into this space.
   l. Choose a yellow as the foreground color and pour it into this button. You should now have two cast members: a red button and a yellow button.
   m. Drag the red button cast member to the middle of the stage.
Now here's the scripting part:

4. Open the Score Window and shorten the red button's sprite to 1 frame.

5. Double-click the scripting channel at the first frame to open the scripting window. The event part of the script is already written for you; all you have to do is fill in the empty line with the command "go to the frame." Close the window to attach the script to this frame. This script will make the frame loop continuously.

6. Make the button swap cast members when clicked on stage.
   a. Click once on the red button's sprite to select it.
   b. Click and hold on the Script Popup Bar.
   c. Choose and release on New Script. This brings up the Scripting Window with the event part of the script already written.
   d. In the blank line type "set the member of sprite 1 to 1."
   e. Set the cursor in front of the first line and press the Return key a couple of times.
   f. In the space you just made type in the following complete lingo script:
      
      on mouseDown
       set the member of sprite 1 to 2
      end
      
   g. Close the window.

Now save the movie and try running it. Granted it doesn't do much-- just changes the button image. But imagine substituting more interesting pictures for the two button cast members (like pictures of an up and a down button). Also imagine attaching some more commands to the lingo scripts you just wrote-- commands that make sounds or that show other pictures. We'll do this in the next tutorial.
This tutorial adds to **Time-Line Handout 12: Sprites as Buttons** so if you haven't completed that one, do so now. You'll also need a few digitized sounds for button clicks. You can either make these in your sound editor or download them from the Multimedia Elements page on our web site. This tutorial will show you how to add a few features to the button you created and how to make it do a several useful things. OK, open up the movie you completed in tutorial #3 and...

1. Add sound to the button:
   a. Use File/Import on the menu strip to locate and import a mouse up sound and a mouse down sound. It really doesn't matter for this tutorial what they sound like, although it probably would be best if they were short and small in file size. You should end up with two new sound cast members as shown here.
   b. Use Window/Script on the menu strip to open the script window.
   c. Use the forward and backward arrows at the top left of this window to move to the button script you wrote in the previous tutorial. (It's the one that begins with "on mouseDown."
   d. Add the two lines indicated by arrows to the script. The command "puppetsound" plays a sound through lingo. That means the sound doesn't have to be present in the score's sound channels-- just in the cast. The first number after "puppetsound" is the sound channel to play it through. The second number is the cast number of the sound. You can also refer to the sound by its name rather than number if you wish. That means that `puppetsound 1, 5` and `puppetsound 1, "MouseDown"` are equal.
   e. Close the script window and run the movie to try your button clicks. Press cmd-. (period) to stop the movie.

2. Add some images to the cast:
   a. Use Window/Paint in the menu strip to open the Paint window.
   b. If the window isn't blank, click on the + button to create an empty space.
   c. Make an image small enough to fit on the right side of stage beside the button.
   d. Repeat steps 2 and 3 three more times to make 4 images total. You should now have 4 new cast members something like those shown here.

Multimedia Authoring - Ti:ME 2B - Teacher Manual
Page 79
3. Make the button switch these new cast members randomly:
   a. Use Window/Script on the menu strip to open the script window again.
   c. Use the forward and backward arrows at the top left of the window to move to the same button script you edited in steps 1c and 1d.
   d. Add the line indicated by the arrow to the mouseUp script.
      ```
      set the member of sprite 2 to random(4)+6
      ```
      This line selects a random cast member between 7 and 10 on each mouse up. Close the window and try it.

4. Make the button look depressed:
   a. In the Paint window, find the yellow button.
   b. Select the Registration tool. A set of crosshairs will appear on the button.
   c. Click just to the left and just slightly above the current center point.
   d. Select any different tool, close the window, and try the movie.

There are a lot of other ways to imply a depressed button: a darker fill color, a thicker outline, a slightly smaller size. You might want to experiment with some of these before moving on to the next interactive gadget.
Interactivity can take many forms in addition to simple button clicks. In this tutorial we'll build a slider that allows the user to input a choice from a continuum of data rather than the simple binary choice that a button allows. Also it will show a good use for Director's Message window. This tutorial adds to Time-Line Handout 13: More Controls so if you haven't completed that one, do it now. If it is completed, open it and...

1. Add a slider to the stage.
   a. In the Paint window, create a vertical line about 2 inches long
   b. Use the + button to open a new work space.
   c. In this new space, create a rectangle about 1/2" wide by 1/4" tall.
   d. Set the registration point of this rectangle at the upper left corner.
   e. Close the paint window.
   f. Open the cast window. You should now have 2 new cast members.
   g. Drag the line cast member onto the stage to the left side of the button.
   h. Drag the rectangle to the bottom of the line as shown in this picture of the stage.
   i. Open the Score window and make the line sprite and the rectangle sprite 1 frame long. Be sure that the line is in channel 3 and the rectangle is in channel 4.
   j. Make the rectangle movable by selecting its sprite (channel 4) in the Score window and then clicking the Movable box at the top of the window.

2. Restrain the slider to a linear path:
   This step will require some program feedback. Here's how to set up this feedback..
   a. In the score, click once on the rectangle sprite (4) to select it.
   b. Click and hold on the Script Popup Bar. Choose "New Script" and release.
   c. In the blank line of the Script window that comes up type: put the locx of sprite 4, the locy of sprite 4
      This command will show the horizontal and vertical locations of the rectangle when the program runs.
      Because the command does not include an "into..." statement, Director assumes that you want the results shown in the Message window.
   d. Close the Script window.
3. Report the slider position to the user.
   a. Open the Tool Palette (Window/Tool Palette in the menu strip). Choose the Field tool.
   b. Draw a small field rectangle on the stage below the button. Click somewhere else on the stage.
   c. Open the Score window and shorten the field sprite to 1 frame. Be sure it's in channel 5.
   d. Close the Score window and open the Cast window. Note the member number of the new field.
   e. Close the Cast window and open the Script window. Navigate to the slider script you made in step 2j.
   f. Just before the "go to the frame" line, add a line that reads: **put 188 - (the locv of sprite 4) into field 14**. (Of course, your numbers will not be the same. Use the ones you entered in step 2j.)
   g. That's it. You're done! Close the window and enjoy running your interactive bells & whistles.
These short example scripts are valuable for more complex user interactions. They should be tailored to your specific needs.

```lingo
on exitFrame
set the loch of sprite 4 to 42
if the locv of sprite 4 > 188 then set the locv of sprite 4 to 188
if the locv of sprite 4 < 40 then set the locv of sprite 4 to 40
set the text of field 10 to 188 - the locv of sprite 4
end

This script restricts the motion of a sprite (perhaps a slider) to vertical only and further restricts it to a set range. The vertical position is placed in a text field on the stage.

global myvariable1, myvariable2
on startmovie
set myvariable1 = 0
set myvariable2 = “Correct!”
exit

It’s often necessary to initialize variables to specific contents (either numbers or text) as the Director movie begins. The global statement makes the contents of the variables available to other scripts in the movie as long as those scripts have identical global statements.

on waitawhile
startTimer
repeat while the timer < 120
nothing
end repeat
end

It is often necessary to pause the playback of a movie or some interaction with the user for a few seconds. Director’s playback head is not accurate enough for many purposes. This script—which counts off 2 seconds in “ticks” (60ths of a second here) — is much more accurate. Notice that the event name “on waitawhile” is a creation of the user. The script can be activated anywhere in the movie just by calling “waitawhile”

on chooseanumber
set mynumber = random(21)+39
set the text of field 10 = “Number = “ & mynumber
end

Games use random functions a lot. This example inserts a random integer between 40 and 60 into the variable “mynumber” and then places the word “Number” and that variable into a text field on the stage.

on startmovie
set the keydownscript to “keyexcluder”
end

Together these two movie level scripts prohibit the user from entering anything other than numbers into a text field

on keyexcluder
if chartonum(the key)>57 or chartonum(the key)<48 then
dontpassevent
end if
end
```

To go beyond these scripts you should analyze and copy those of other Director authors.
These templates are available online as both shockwave files and source code.

**Loudness Matcher**

Demo: http://music.utsa.edu/TIME-MM/dcr/LoudMatcher.dcr

Source: http://music.utsa.edu/TIME-MM/sitzip/LoudMatcher.dir.sit

This is a small, movie that uses buttons, text fields, and a slider as interactive gadgets. Its single function is to test the user's ability to match a given tone's loudness. The user clicks the left button to generate a 1 second tone with a randomly selected intensity. After the tone stops, he tries to match its loudness with a similar tone whose intensity is controlled by the slider. To check his accuracy, he clicks the right button. (Created by D.S.)

**Magnifier**

Demo: http://music.utsa.edu/TIME-MM/dcr/Magnifier.dcr

Source: http://music.utsa.edu/TIME-MM/sitzip/Magnifier.dir.sit

Magnifier is a movie designed to demonstrate Director's ink effects. Drag the "magnifying glass" around the image to see the effect. (Created by D.S.)
Computer Number Converter
Demo: http://music.utsa.edu/TIME-MM/dcr/NumberConversion.dcr
Source: http://music.utsa.edu/TIME-MM/sitzip/NumberConversion.dir.sit

This movie is used in web-based Computer Applications in Music course to give students some hands-on experience with how computers interpret numbers. The green strip represents 8 timed segments of electrical pressure (a byte) which is really all that a computer deals with at its most basic level. Clicking these segments on or off is translated into binary math, decimal math, hexadecimal math, ASCII code, and MIDI message. The mathematical algorithms and lingo underlying the program are quite simple and easy to duplicate. (Created by D.S.)

Sound Hound
Demo: http://music.utsa.edu/TIME-MM/dcr/SoundHound.dcr
Source: http://music.utsa.edu/TIME-MM/sitzip/SoundHound.dir.sit

This is a little game designed to Demonstrate the accuracy of a student's loudness perception. Press the "Start Sound" button, move the cursor (the red ball) around the game board until the sound is loudest, and then click. The game will reveal whether or not you really selected the area of loudest sound. (Created by D.S.)
**Slide Show**

Demo: http://music.utsa.edu/TIME-MM/dcr/slideshow.dcr  
Source: http://music.utsa.edu/TIME-MM/sitzip/slideshow.dir.sit

A movie designed to Demonstrate both linear control (the triangles) and non-linear control (the button strip) in Director. Because it uses photographs, it's larger--about 300k--and so will take several minutes to download. (Created by D.S.)

---

**MIDI Sequencer**

Demo: http://music.utsa.edu/TIME-MM/dcr/MIDIKeyboard.dcr  
Source: http://music.utsa.edu/TIME-MM/sitzip/MIDIKeyboard.dir.sit  
(requires Yamaha MidXtra)

The world's cheapest MIDI setup! This movie was designed to illustrate how a sequencer uses numeric codes to create a musical performance. Notes and durations can be entered from the keyboard. Any MIDI command can be typed into the "sequencer" at the top of the screen if the user knows how to code it. (Created by D.S.) As far as the multimedia goes, it Demonstrates several novel types of buttons (the keyboard is actually only 2 buttons) and two types of text fields.
Melody Maker
Demo: http://music.utsa.edu/TIME-MM/dcr/melodymaker.dcr
Source: http://music.utsa.edu/TIME-MM/sitzip/melodymaker.dir.sit
(requires Yamaha MidXtra)

This is an attempt to Demonstrate some principles of algorithmic music composition using Director. The interface is simple-- each button controls one "rule" of a good melody. The mathematical algorithms generating the melody are not very complicated either.

Real Time 3D Manipulation
Demo: http://music.utsa.edu/TIME-MM/dcr/RCAplug.dcr
Source: http://music.utsa.edu/TIME-MM/sitzip/RCAplug.dir.sit
(requires QuickDraw 3D Xtra)

A student can view and manipulate common objects in a virtual 3D space using this movie. Here the mouse controls the orientation of the RCA plug in real time.
Multimedia authoring, the National Standards and the TI:ME Strategies for Music Education

1. Which national standard(s) can be addressed using multimedia authoring software?
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________

2. What specific ways could multimedia authoring be used to address these standards?
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________

MENC Standards:
1. Singing, alone and with others, a varied repertoire of music.
2. Performing on instruments, alone and with others, a varied repertoire of music.
3. Improvising melodies, harmonies, and accompaniments.
5. Reading and notating music.
6. Listening to, analyzing and describing music.
7. Evaluating music and music performances.
8. Understanding relationships between music, the other arts, and disciplines outside the arts.
9. Understanding Music in Relation to History and Culture.

Now review the Teaching Strategies listed in the TI:ME technology strategies document. See Appendix A of the Technology Strategies for Music Education. Describe one or more ways to use multimedia authoring in your teaching position:

[TIME INSTRUCTORS MAY PHOTOCOPY AND DISTRIBUTE APPENDIX A IF NECESSARY]

TI:ME Tech. Strategy #  Teaching Application
___ __________________________
___ __________________________
___ __________________________
___ __________________________

List three ways that multimedia authoring could be used in your teaching position:
1. ________________________________________________________________
2. ________________________________________________________________
3. ________________________________________________________________
Those who download files from the Internet and who create multimedia presentations and lessons should understand the copyright law. The right of creative people to control the display and circulation of their work is assured by law. As a general guideline, however, a copyright notice must be displayed on copyrighted material. If no notice is displayed, the material may be used but out of courtesy, permission should still be requested. If a copyright notice is displayed, then permission for using the material must be obtained from the copyright owner before using the material. On the Internet copyright permissions are easily obtained using e-mail. All materials not created by the author should contain a complete citation giving credit to the source.

For further information, refer to:

**The United States Copyright Law - A Guide for Music Educators**

http://www.menc.org/copy/copyr.html

Comments:
Appendix 1

Multimedia Authoring

Multimedia authoring software is generally in rapid transition and many products come and go as the result of corporate purchases and takeovers. Furthermore, the features of these products vary with time. TI:ME recommends the use of currently available, cross-platform software. Instructors should confirm that these programs meet this criteria.

PRESENTATION PROGRAMS

PowerPoint - Microsoft  http://www.microsoft.com/powerpoint/
Astound - Astound  http://www.astound.com

STACK-BASED AUTHORING PROGRAMS

HyperStudio - Roger Wagner Publishing  http://www.hyperstudio.com/
SuperLink - Alchemedia  http://www.alchemediainc.com/

TIME-LINE AUTHORING PROGRAMS

Director - Macromedia  http://www.macromedia.com/software/director/
Apple Media Tool - Apple Computer  http://amt.apple.com/
mTropolis - Quark  http://www.quark.com/about/itm056.html