Multimedia – A Guide

Topic 1 – Macromedia Director

What is Macromedia Director?

Director can create everything from the simplest PowerPoint-type presentations to the most complicated Kiosk databases. The main variable that will determine what you can accomplish with Director is your own skill, education level, and willingness to learn. In this very abbreviated format I will try to introduce you to some basic Director capabilities, basic set-up stages, file formats, sound settings, and video settings.

Where can I learn more?

The Internet is your best friend when you have a problem that you need to solve. You can find the entire lingo (Director's language) dictionary online, and there are many resources available to you.

http://	**My 2 favorite Online Resources:**
	http://www.macromedia.com/support/director/
	http://www.director-online.com/

Where do I begin?

Gather information

Beginning a Director presentation of any size can be a daunting task. The most important thing is to have as much information about what needs to be accomplished for the project. From these guidelines you can begin to map out your presentation.

Here are a few sample questions that you will need to know:

- Where and how is this project to be displayed and/or distributed. It's a good idea to know your base system requirements at this stage. You can discuss display size, memory, etc. with your IT person.
- How much content is expected and how will you receive it?
- Which multimedia elements are expected to be included (video, sound, video with sound, Flash, etc.)?

Map it!

When you have your information gathered, you should build a map of how you expect the presentation to work. This map could be very simple to start, but it's a good idea to add to it as you receive more content. As you are drawing the map (use Inspiration or Visio for professional site maps that are easy to modify), keep in mind how you are going to make it work. Make notes for yourself on possible solutions as you go and keep a list of potential problems to research. These maps also make a nice visual aid to present to your client or manager to show the basic operation of the presentation.

Design

Now you have a plan. It is much easier to tailor your interface to your plan than to tailor your plan to your interface! Here are some Design Tips to keep in mind:

- When designing an interface for a presentation, keep your user in your mind. Complex, artistic navigation looks cool but can be cumbersome to use.
- Try to keep the scheme of the navigation simple and easy to understand.
- Design your interface in PhotoShop and keep in mind button states.

  – the up (no mouse), mouse over, mouse leave, mouse down and mouse up (click)) as you go.

- Usually you will have a home page with your creative button scheme and an inside page that allows the user to move easily through your presentation. Once you have a sign-off on your design you can...

Make a mock-up

A mock-up of your presentation shows the actual functionality of the presentation with "dummy" content. This stage can be critical in identifying any problems you will need to work out for your plan. Try to include "dummy" content in all categories that you are expecting (sound, video, navigation, etc.).

Set-up

When you set up your movies in Director, keep these points in mind:

- Switch your monitor to Thousands of Colors. This will automatically make your presentation thousands of colors. If you design in millions of colors, some of your users may not be able to view your presentation or the color will be severely altered.

- Presentations that you know will be used on better computers can be set up at 800 x 600 points.
  - You may need to reconsider if you have any full-screen video or an extreme amount of content.

- Presentations for the general public or ones that will be seen on older machines should be set up at 640 x 480 points.

- Pay attention to how your navigation is going to work. It may be a good idea to break each section or chapter into its own Director movie. This will decrease load time and will allow your presentation to run faster.

Director structure

Some terms you'll need to know for the following sections:

stage – Just like in theater, this is where everything happens. Where all your sprites interact and what your end-user sees when experiencing your presentation.

frame – A space in time, based on the frames per second (fps) you apply to your movie (set in the tempo channel or with lingo).

playhead – This indicates the frame you are on during movie playback or editing.

score – The score is the timeline of the presentation. This is where you control your sprites and their interaction. The score is made up of several different channels for controlling different elements of the movie. These elements are the tempo channel, the color palette channel, the transition channel, 2 sound channels, the script channel, and the sprite channels.

cast – The cast consists of your available "players" for use in your presentation. Any object, sound or video is held in a cast library.

cast window – Also cast library where all graphics, text, and behaviors are stored for use in the movie. As you build your movie and things change, delete or replace old members. There is no "delete unused" in Director.

sprite – Once a cast member is placed on the stage, it becomes a sprite (an instance of the original cast member) in a sprite channel. Tweening, ink effects, and behaviors are applied to sprites.

sprite channel – Channels 1–150 on the score. You are limited to 1,000 channels, but the number of frames is practically infinite. New frames and channels will be automatically added, or you can add them manually.

tween – Moving an object from point A to point B.

script – Lingo script.

effects channels – The top collapsible channels in the score are the effects channels, which in order from the top are the tempo channel, the palette channel, the transition channel, 2 sound channels, and the script channel.

tempo channel – The place where you can set timing, delay, and wait statements associated with your movie.

palette channel – Contains color-palette information, including color transition effects you use in your movie. If you do not specify a color palette, you will use the default palette, which would be fine unless you want to specifically use only a palette of colors you create.

transition channel – The place where you can use transition cast members to control the visual transition from one frame to the next in your movie. These are very PowerPoint-like; you specify the type and time and Director will do your transition for you.

sound channel – You can place sound directly in the sound channel. This will limit you to 2 sounds at any one time. A better alternative is to use lingo where you can have up to 8 sounds at once (Director mixes and sends them to the sound card for you).

script channel – Where you put your lingo script affecting anything but an individual instance of a cast member on the stage (a sprite). For instance, button lingo goes on the button itself, while lingo that controls the playhead goes in the script channel.

lingo – The very powerful scripting tool used by Director. All things can be controlled in your movie with lingo, depending on how advanced your knowledge and code become.

Add content

When you actually add content to your presentation, you aren't working in a real WYSIWYG environment. Usually your graphics come from one program and your text from another. I recommend you have your content proofed and revised before you begin translating it to multimedia formats. This will save you and your proofer headaches down the line by working with truly finalized content.

How do I do that?

Adding content can be tricky. It's important to think it through before you charge blindly into PhotoShop and create everything all at once. Because you have an idea of how your presentation works, you may have already experimented with navigation schemes and content types for your dummy. Keep in mind the file format that works best with the different content types. Here is a list to give you an idea:

File formats

.pct – These files are your basic graphic format. You save 16-bit picts (set your monitor to Thousands of Colors first) from Photoshop (or FireWorks if you are so inclined) the size of the stage in Director. These files are imported into Director, and extraneous information is cut off the cast member (more on that later).

.png – These are great little files for objects that need to move but do not have a hard edge. A png can be exported (use "save for web" in the file menu) with a transparent background. It is best to slice these images then reposition them in Director. While this is more difficult, your file sizes will be smaller (more on this later).

.swf – A file created by Flash. Shockwave files (swf) are great for graphic cast members, animated cast members, text, and even navigation elements. You do have limited ability to control lingo with Flash's own ActionScript. Text looks best as a swf because it's a vector-based format. Text exported as png or pct can be used as headlines or larger things, but body text is best left to swf.

.wav – A sound format common to Windows. You can use wav files with no extras or worries. You can also add sound markers into wav files (using a sound-editing program). This format will work for MAC (with Quicktime installed) or PC.

.aif – A sound format for Mac or PC. You can add sound markers to aiff's (using a sound editing program).

.mov – A QuickTime movie format for Mac or PC. You must include QuickTime (and check for QuickTime) to be sure the user has it installed. Most Macs will have it; not as many PCs will have it installed or updated. You can also put your sound into this video format (recommended).

.avi – Microsoft digital video format. No extensions are needed to view avi on a PC. You can also put your sound into this format (recommended).

.mpg – PC video format (Macs can see, but DirectMedia is not a Mac extra, more on that later). Using mpgs enables the files to support a high level of compression and will probably be your smallest format — and your biggest headache. You can put your sound into this format also.

When you are creating content, it is very important to be aware of not only how this content will look, but how it will function. If you need an object to move, say as the introduction builds on screen, a pct file may not be your best solution. Pct files cannot have a background. You can use an ink effect in Director to force a background color to be removed, but this will usually leave a pixilated edge around your art. There are ways

to make this less noticeable, but most are pretty labor intensive. Instead, consider exporting a png with a transparent background; or if your shape is purely vector, consider a swf single-framed character or a multiframe Flash movie.

Make your content work for you

But first, a few more definitions:

property inspector – A tool that automatically changes global elements context to display and sets properties for how elements are displayed in a Director movie.

behavior inspector – A tool that allows to to create custom behaviors from existing predefined behaviors.

paint window – Used to create and edit bitmap graphics within Director.

behavior – A special cast member that defines operations or procedures.

event – An activity usually by the user or movie, such as a click or the start of the movie.

action – The consequence to the event.

Export

Now you've got all your content created, you have background and static elements in PhotoShop created to your stage size at 72 dpi (I usually create my background at a higher resolution because you may need those elements for a CD case or other collateral to accompany the multimedia piece) and you've exported them as a pct.

To export as a pct, follow these instructions:

- Double-check that your monitor is set to Thousands of Colors (or 16-bit color).
- Turn on all background layers behind the static object you wish to export in PhotoShop.
- Go to File > SaveAs and choose .pct
- select 16-bit and click "ok"

To create and export your Flash text or graphic members:

- An easy way to get the layout of your text is to export a jpg of the background images and import that into Flash to work on top of, this gives you a better idea of placement without going back and forth between Flash and Director as much.
- Type in your text and change the color and layout to the way you wish it to appear in the final Director piece; create a symbol.
  - Go to Insert > Convert to Symbol... this will store the text in your Flash library so future edits can be completed a little quicker.
- Create all your text and graphic elements, making each a symbol for future use.
- Now place one symbol on the Flash stage, modify the stage size to be a few points larger than the symbol (go to Modify > Document)
  - Removing dead space around the Flash character saves you a little on file size.
- Export the swf by going to File > Publish Settings
  - Check swf only in the Formats tab
  - In the Flash tab, change the drop-down menu to Flash 5; you cannot import Flash MX (or Flash 6) documents into Director 8.5
- Repeat this procedure for all text and graphic elements; save the Flash file in your artwork folder for future reference.

To export your png elements, bitmap elements with transparent backgrounds:

- Double-check that your monitor is set to Thousands of Colors (or 16-bit color).
- Turn **off** all background layers behind the static object you wish to export in PhotoShop.
- Use the slice tool (looks like an X-acto knife) and put a box around the object (and any shadow) you wish to export.

- Go to File > Save For Web and select the slice(s) with the slice tool (left margin)
- Choose png-24 from the drop-down menu to the right
- Check the transparency box
- Click "ok"
- Change Format to "Images Only"
- Change "All Slices" to "Selected Slices"
- Name the slice with your predefined clever naming convention
- Click "ok"

You have all your text as it will appear in your final piece created to size and exported from Flash, all static elements exported as 16-bit pct images, and any non-Flash elements you need transparent or movable you've exported as a transparent background png.

Import

Now you can open Director and set your stage to the proper size and background color (when you make your executable, your background color will be the color that surrounds your presentation). You do this by using the movie tab of the Property Inspector palette. To bring your cast members into the cast, go to File > Import. I usually import same-member types together, all the pngs, all the swfs, and all the pcts. When you choose a file (or multiple files by using the "add" feature of the import dialog) to import, a dialog box will appear with options. Bring everything in at 16-bit except pngs, which MUST be 32-bit color. You can check the box that says to use the same settings for the remaining images. Now if you open your cast palette you will see all your imported members.

Trim and prep images

Each member type is treated differently. Of the pct files you made, you will have exported them on the background of the piece and you now have to trim them down. For example, if you made your background and your 3 navigation tabs pct files (and states of the navigation buttons), they are now all the size of the stage and if you put them into sprite channels, they will cover each other up. To trim the pcts, you double-click on the cast member (in the cast window) and the paint window will open with your image in it. To trim, simply use the rectangle select tool and remove as much as possible from the image. If your button is only in the far right corner of the background, trim off all the data EXCEPT the actual part of the page that will act as the button. Do the same thing for all the states of the pct buttons you've imported.

Put on stage

When you place cast members on the stage, they go into the sprite channels. The layering of the sprite channels works exactly in reverse of PhotoShop and other image manipulation programs. Channel 1 is the very bottom, and all things go on top of it. So place your background pct on the stage. You will see that Director automatically places the image in the lowest available sprite channel. Now if you look at the Property Inspector palette, you will see it has changed to show all available properties of the selected sprite. You can center your image by using the X and Y coordinates of the sprite (all coordinates are figured from the center of the sprite on stage, so put in half of each dimension of your movie to get the object in the center). You can also drag a cast member directly into a sprite channel. It will automatically be centered on the stage. Keep your layering in mind as you begin to place things on the stage. For instance, if you do not want anything to tween on stage over your navigation buttons, place them in a higher sprite channel so there are empty channels available in which to add more content. You can always move objects to different channels, but taking the 2 seconds to think about it now will save you the headache of moving it later. Place all "up" states of objects on stage and make sure all other button states are in your cast library.

Make a button work

Well here we are — our first shot at Lingo. Here is a basic example that you can embellish and change to your liking. But this will show you how the behaviors work. Button states are easy; you can use a type of auto-code that Director has built in for you.

First you may want to create a different cast in which to keep your behaviors.

- Open the cast window and click the top left box
- Click and hold, select "New Cast..." from the pop-up window
- Name it 'behaviors'

Now, with your new behavior cast open, select the up state of a sprite on the stage to make a button.

- Open the Behavior Inspector Palette
- Click the top left button, the big plus symbol and select "New behavior..." from the pop-up window.

    Try to use a consistent naming convention that you can recognize easily, something like "btn1_toMain" or however you will remember what the behavior is.

- Click the plus sign next to Events and choose "Mouse Enter" from the pop-up window.

    – when the user's mouse enters the area of the sprite, a sprite event is activated.

- Next choose an action for this event, click the plus sign next to Actions and choose "Sprite," then "Change Cast Member..." from the pop-up window.

- A dialog box will come up; choose the cast member that you want to be the "over" state of the button. With your clever naming convention, this will be a cinch in the pop-up list!

- Next add another Action to the "Mouse Enter" event, click the Action plus sign again, and choose "Cursor," then "Change Cursor..." and choose "Finger" from the pop-up list.

    – This list contains all system level cursors you can use

    – This will change the cursor to the way a user is accustomed to seeing a button function.

Your button will change its state when a user rolls over it. But it won't change back, so let's add a new event.

- Click the plus sign next to Events and select "Mouse Leave" from the pop-up window.

    – When the user's mouse leaves thearea of the sprite, a sprite event is activated.

- Click the plus sign next to Actions and choose "Sprite," then "Change Cast Member..." from the pop-up list.

- Choose the up state of the button from the pop-up list.

- Now add the cursor change, add another action to mouseLeave, and choose "Cursor," then "Restore Cursor" to change the cursor back to the arrow.

Now your button will function like a button — it just doesn't do anything. If you would like, you can also add a Mouse Down event if you made a down state on your button. We will put your action into the Mouse Up event.

- Add a Mouse Up event and the action will be "Navigation" then "Go to Marker..."

Once you have markers, you can select the proper one from the pop-up list. You can add button functionality to test appearance then go back into the behavior and add the actual navigation actions later when your movie is more complete.

Structure your file

Now you have all your cast members imported, trimmed, and ready to go. Now it's time to build the movie to time. You can begin by dragging your members on and adding tweens, effects, etc. to make the movie. The main way your playhead navigates is by commands to stop and go to different places.

- On frame 1, set your tempo in the tempo channel; usually 12-15 frames per second (fps) is good.

    – Now you can judge the time span in which you want things to happen.

    – If you want a 5-second tween and your movie tempo is 15 fps, that's simply 5 (seconds) x 15 fps = 75 frames in the tween. Easy, right?

Now, how to stop and add these commands and places for the playhead to go. To stop the playhead:

- Open your behavior cast
- Double-click in the script channel on the frame on which you want the playhead to stop
- Enter the following script;

<table>
<tr><td>lingo</td><td>

```
on exitFrame me
   go the frame
end
```

</td></tr>
</table>

Be sure to name your script in the behavior script window for future reference; you'll be reusing this behavior a lot. To have a place for the playhead to go, you use a marker. To add a marker to your movie, put the playhead where you want a marker to be, for example, before an animation or tween to bring information on stage. Simply click in the white area above the tempo channel and a marker will be added. Be sure to name it something easy to recognize — remember you'll be using these names from pop-up lists so you need to be able to recognize them easily. Now you can add the basic structure of your movie. Depending on the complexity of your content, you may need to break your movie up into several movies, maybe by chapter or category. You can still navigate between them in the same way; it's just a few extra lines of script.

This script tells the playback head to go to the marker named start:

<table>
<tr><td>lingo</td><td>

```
on exitFrame me
   go to "start"
end
```

</td></tr>
</table>

This script tells the playback head to go to the marker named question in the movie chapter2

<table>
<tr><td>lingo</td><td>

```
on exitFrame me
   go frame ("question") of movie ("chapter2")
end
```

</td></tr>
</table>

Digital video

As we saw earlier, there are several options for the format of your digital video elements. Deciding which type of video to use is going to be determined by where your presentation is going to be viewed and what kind of software options are available. For instance, if the planets have suddenly aligned and the world has become enlightened, you are creating a Mac-only presentation and can go with Quicktime. As the planets are still spiraling toward a PC-based hell, you'll probably use an avi or mpg. If you have a cross-platform CD you are producing, you may consider adding a QuickTime installer, with a check to make sure the user has the same version or better that you are using. If you are going PC only, you can use a Director extension called DirectMedia to import all your mpg movies or simply use an avi. Avi's are simpler to use, but mpg's look much better. But first, let's back up a little.

How do I make a movie?

You may have digital video shot from a camera, or a client may send you a video file. The biggest hurdles in digital video are file size and compression. If you are working from the source video, you may have to edit the video, add effects, sound, transitions, text, etc. To edit pieces of video, stills, and sound, you can use Adobe Premiere. If you need to add text or animate any object, you will need to use Adobe AfterEffects. You should export your video at the highest quality you can as a QuickTime video.

Video Processing Information
• Example #1: Test file is 109MB with no adjustments to file.

   @2400 Data Rate the size ended up   5MB = 4.5 %

   @4800   10MB = 9.0 %

   @6400   12.5 MB = 11.5 %
• The higher the data rate the less time it has to render.

   – Example #2: A 550MB file @ 6400 Data Rate took only an hour
• Example #3: A 40 min capture on high setting took up 10.25 GB of space.

   – When you have a large capture, approximately 10Gb, save it out in QuickTime to make it smaller so it has an easier time in Cleaner. Larger files like 10GB end up erroring in Cleaner.

- Example #4: AfterEffects
  - On the final render, make sure the resolution setting is at full resolution
- Example #5
  - Capturing video and exporting it out to "video" is more of a workable size than exporting it as "animation," which is superior quality, but is 1,100% larger (MB) when saved out at Millions+ of Colors (500% larger when saved out at thousands). Both need to be saved out at high quality.
- Example #6
  - The key is to capture video, add (.mov) so you can import it into AfterEffects or Premiere. For either, export as animation set at high quality, thousands of colors. This is fine for video on a CD.

Once your export is complete (which could take quite a bit of time) you need to run your videos through Cleaner. In Cleaner, you make them into the format and size you need in your presentation. You can also add fades at the beginning and the end, adjust color and contrast slightly, and do a few other simple modifications. I recommend you do all your videos at once with the same settings. But just in case, you should also make a note of what settings you used — the client may make a change and you will have to re-export a video in the future. There are too many variables to list specific settings for Cleaner here. A few pieces of advice:

- If your video is going into PowerPoint, it must be an mpg-1.
- If you are exporting an mpg-1 for a video CD, be sure you check audio even if there isn't any audio. DVD players need there to be an audio track, even if it's empty.
- Try to use the best quality you can afford size-wise.
- For PC presentations, mpgs look the best, but avi's, though not as high quality, have no extension or system conflicts for the user.
- All videos can use the animation compressor and Millions+ of Colors for super-duper quality, but the video compressor and Thousands of Colors is a good alternative if you have large files and want to keep your size down. Typically for most CD projects, Thousands of Colors is fine.

Known mpg conflicts
Occasionally with video you will get a black flash as the video begins. If you are using DirectMedia this is a known problem and there are several possible solutions.

First, the most practiced way of preventing the dreaded blink at the beginning of a digital video.

- Position your mpg, mov, or avi on the stage where you want it to play.
- using the Property Inspector add 1500 to it's x (horizontal) coordinate
- Place the following code in the script channel at the first frame of the media sprite

```
lingo        on enterFrame me
               sprite(13).movieRate = 1
               sprite(13).LocH = sprite(13).LocH - 1500
             end
```

  - this code first sets the property of sprite 13 (or whichever sprite channel your digital video is in) to 1 or play, 0 is stop
  - then the location of sprite 13 is moved -1500 pts so it will be back on stage where you originally placed the cast member. If you zoom down you will see it stuck out in right field
- This code basically masks the flash by having the video kick on off-stage.
  - press play to test

Second, the problem with DirectMedia. Tech support acknowledges that the black flash is a charateristic of using the hardware acceleration features of DirectX, leaving an artifact (black flash) on startup and finish when the video overlay window redraws.

They offer this solution:

lingo

Documentation from Tabulerio (http://xtras.tabuleiro.com/)

How to remove the black flash that happens on some Windows machines when MPEG files are loaded and unloaded

ID: MA006   PRODUCT: MPEG ADVANCE XTRA   PLATFORM: WIN   VERSION: 1.03

In some Windows machines there is a black flash frame that appears at the beginning and the end of MPEG playback, in DTS mode. This black flash is a characteristic of using the hardware acceleration features of DirectX, and occurs when an YUV overlay surface is being created by your video driver, in order to optimize playback using hardware accelerated features in your video card.

One way to remove this flash is to reduce hardware acceleration features in the video driver, but this is not practical or desireable since it affects performance. However it should be possible to hide the flash by delaying the exhibition of the video until content has been rendered.

In MPEG Advance Xtra a developer can hide this artifact by using the visible of sprite property. Just drop the following behavior over the video sprite, and it will take care of changing the visible of sprite at startup and end of sprite spans, hiding possible black flashes caused by redraws of the video overlay window.

```
property pHidden
on beginsprite me
sprite(me.spritenum).visible = false
pHidden = true
end


on endsprite me
sprite(me.spritenum).visible = false
--the next two lines are required for Director to update the stage
the stagecolor = the stagecolor
updatestage
end


on prepareFrame me
if pHidden then
if sprite(me.spritenum).currenttime > 0 then
pHidden = false
sprite(me.spritenum).visible = true
end if
end if
end
```

A third solution that I tried was to start the movie off stage, add a marker in the mpg, at that marker shoot the movie on-stage and add another marker almost before the end and shoot the movie off-stage. This had limited results, and limited testing. It maybe a possible solution with further testing and adjusting.

lingo

```
on exitFrame me
  if isPastCuePoint(sprite 14, 1)=TRUE then
    go to the frame + 1
  else
    go the frame
  end if
end
```

– this code looks for the cue point you added in the DirectMedia window of the video spirte and decides if the cue piont had been passed or not, if it has it advances 1 frame where you have your code to shoot the video onstage. otherwise (else) the playhead will wait on that frame until the condition is true.

This is your first if statement. If statements are very powerful tools in lingo.

A fourth solution is to set up the mpg to pause at start, shoot it on stage, then use lingo to play the mpg. This had limited testing and the results were not conclusive.

lingo
```
property videoSprite --the sprite number that is getting controlled
property videoCommand -- the command to be issued


on exitFrame me
  videoSprite = 14
    videoplay(sprite the videoSprite of me)
    go to the frame + 1
end
```

– this code uses properties to make its decision. basically it says play sprite 14 (or whichever channel your video resides in) and advance the playhead 1 frame.

Making video function correctly in your presentation is a huge challenge. Each video has so many variables that affect its performance you have to use your past experience as a astarting point and solve the problems as they happen. The best advice is to keep the video in your mind and realize it's not something your going to be able to drop in and be done with. The video processing and intigration will more than likely take longer than you anticipated and present some new and unique issues. Use your support devices on the internet and the documentation in the multimedia books.

Audio

Adding audio to your presentation can make it more dynamic and interesting. Audio can also distract the user from the message or content of your presentation. Overkill on audio beeping, swishing, etc can overshadow your content, so be careful of overdoing it.

Director provides you with 2 sound channels to use for your sound. But with lingo you can add up to 7 separate sounds and Director will

If you are using sound with video, like for a voice over, I recommend you put your sound in your video using Premiere or AfterEffects. Loading sound and video could cause some syncing problems. If you have animation with sound you can use markers and lingo to move your animation to sync up with your sound.

To create your sound you probably have huge aif or wav file from the voice over talent or music from a sound clip art CD. It is advisable to break your sound up so it's not one big long sound file. The best way to plan this is with your storyboard and your script. Read through your script and mark where you think the best places will be to break up the sound into pieces. While you're doing this you can also plan your markers in your sound. Or where you need the playhead to wait for an animation to start or wait on a particular screen. I usually draw a bar in and write the name of the marker on the script (pencil may be a good idea here).

To break the large aif (or wav) into manageable pieces:

• Keep the original untouched, you may need to refer to it later

• Open the file in Sound Studio (or your prefered sound editing program)

• To cut the file select the portion of the sound you want and copy and cut it out of the big section

• Go to file > new and select 22.050 kHz, 16-bits, mono and paste in the data

• You can delete excessive silence at the begining and end of the file by selecting it and hitting delete.

• If you need the sound quieter you can select all and go to Filter > Volume and lower the dB

To add markers to your sound file

• Put the cursor (playhead) where you want the marker to be and go to Insert > Marker

• To name the marker click on the default Marker 1 and type a new name

• You can move markers like a guide in PhotoShop

Export the sound

- Export all sounds for multimedia as 22.050 kHz, 16-bits, mono
- If you have a very small, simple sound like a beep you may be able to go lower on the kHz size.
- In Sound Studio go to File > SaveAs and save the sound as an aif file to keep the markers.
- To export sounds from QuickTime Player go to File > Export
    - choose wav and click Options
    - click Settings under sound and change the settings to 22.050 kHz, 16-bits, mono

Sound and Lingo

Using lingo to control your sound will make your sound easier to control.

Import the sound into Director. You may want to build a new cast to keep your sounds in, if you have several sounds that are large in size an external cast will help you out.

external cast – Cast members stored ina separate file that can be shared between movies or used in libraries of cast elements.

Building and external cast for sound and video elements will enable you to keep all your eggs in one basket so to say. If you have several Director movies, say for each chapter of your presentation, to update sound you would have to go into each movie and replace the appropriate members. With an external cast you update them once and then each Director movie will pull the updated member.

To make an external cast

- Open the cast window and click the top left box
- Click and hold, select "New Cast..." from the pop-up window
- Name it 'sound.cst'
    - it is important to name files with extensions for the PC
- click external and leave "use in current movie" checked
- You will now have a file on the same level as your Director movie called "sound.cst" you must always have this file on the same level as your director movie

To link an external cast

- If you make a new Director file and need to use elements from your external cast.
- Go to Modify > Movie > Casts...
- Click link and select your previously created external cast.
- Now your content in your cast will be available in the current Director movie. Repeat as needed.

Puppet sound is the easiest way to add sound using Lingo.

lingo	on exitFrame me puppetsound 1, "soundName" end

- This script will put the sound in channel 1. You can use channels 1 – 7 with lingo. Do not use the same channel simultaneously.

You can control the timeline according to your sounds with lingo also. You can reference your Cue Point names by number (first, second, third cue point) or by name in your script.

- to find cuePoint name use: put the cuePointNames of member "SOUND NAME HERE" into the message window in Director. (Window > Message)

| lingo | ```
on exitFrame me
  if sound(1). isPastCuePoint(1)=TRUE then
    go the frame + 1
  else
    go the frame
  end if
end
``` |

– In this script your playhead will wait until the sound has passed the first cue point. You could also put the name of the cue point in "" within the parenthesis.

There are lots of ways to control sound with lingo. For further information and examples please reference the hard copy support materials and the online resources.

Director Library

Director graciously supplies us with a library of ready-written script with drag-and-drop functionality. There are several behaviors that you simply drag onto the sprite and a dialog box will come up where you fill in your variables and the code is complete.