Sound and Video

Topics

- Sound
  - Timeline
  - ActionScript
  - Sound Design for Instruction
- Video
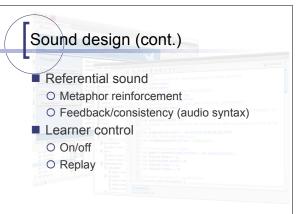 - FLVPlayback Component
  - Adobe Media Encoder

Sound design (Bishop & Cates, 2001)

- Music
  - Set mood/control pacing
  - Keep short
- Narration
  - Text <= narration
- Environmental sounds
  - Inverse length relationships
  - First person (immersion)
  - Feel free to exagerate

Sound design (cont.)

- Referential sound
  - Metaphor reinforcement
  - Feedback/consistency (audio syntax)
- Learner control
  - On/off
  - Replay

Sound formats supported by Flash

Type	PC	Mac
.mp3	x	x
.wav*	x	x
.aiff*	x	x
.au	x	x
.midi		
.qta or .mov (quicktime)	x	x

Impacts on file size

Sample Rate (kHz)

Sample Size (bits)

Sound compression in Flash

Compression	Use
ADPCM	Good for short sound effects (correct/incorrect response, button clicks, etc . . .)
MP3	Ideal for music, best general purpose choice
Speech	Great for voice only
RAW	No compression = huge file sizes, but best quality

Adapted from CS3 help file

Issues with sound

- Action Script is preferred control mechanism.
- Event vs. Stream

Sound in the timeline

- Effects (mostly custom effects)
- Export settings/compression (true of sounds in timeline or sounds controlled with actionscript)

Sound using ActionScript 3.0

- Embedded sounds
  - Linkage names
- Custom class
- Sound class
- SoundChannel class
- SoundTransform class
- Alternative:
  - Custom Sound class

Things to know about sound

- Limited to 32 sound channels
- If importing at run time, has to be .mp3
- Embedded can be .mp3, .wav, .aiff . . .
- Still worth the effort.
- Sandbox issues

Sound using ActionScript 3.0

- Example (simple)

```
// create a new instance of an embedded sound
// that uses the linkage name "quietlySnd" (this is
// your custom class)
var backMusic:quietlySnd = new quietlySnd();

//  if you're happy with the sound as is:
backMusic.play();
```
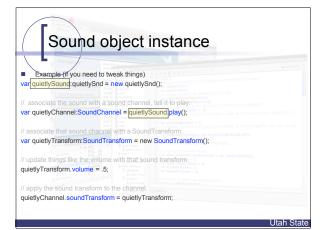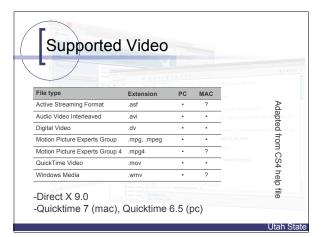
Sound using ActionScript 3.0

■ Example (if you need to tweak things)

```
var quietlySound:quietlySnd = new quietlySnd();

// associate the sound with a sound channel, tell it to play:
var quietlyChannel:SoundChannel = quietlySound.play();

// associate that sound channel with a SoundTransform.
var quietlyTransform:SoundTransform = new SoundTransform();

// update things like the volume with that sound transform.
quietlyTransform.volume = .5;

// apply the sound transform to the channel.
quietlyChannel.soundTransform = quietlyTransform;
```

Sound object (linkage name)

■ Example (if you need to tweak things)

```
var quietlySound:quietlySnd = new quietlySnd();

// associate the sound with a sound channel, tell it to play:
var quietlyChannel:SoundChannel = quietlySound.play();

// associate that sound channel with a SoundTransform.
var quietlyTransform:SoundTransform = new SoundTransform();

// update things like the volume with that sound transform.
quietlyTransform.volume = .5;

// apply the sound transform to the channel.
quietlyChannel.soundTransform = quietlyTransform;
```

Sound object instance

■ Example (if you need to tweak things)

```
var quietlySound:quietlySnd = new quietlySnd();

// associate the sound with a sound channel, tell it to play:
var quietlyChannel:SoundChannel = quietlySound.play();

// associate that sound channel with a SoundTransform.
var quietlyTransform:SoundTransform = new SoundTransform();

// update things like the volume with that sound transform.
quietlyTransform.volume = .5;

// apply the sound transform to the channel.
quietlyChannel.soundTransform = quietlyTransform;
```

Sound Channel instance

■ Example (if you need to tweak things)

```
var quietlySound:quietlySnd = new quietlySnd();

// associate the sound with a sound channel, tell it to play:
var quietlyChannel:SoundChannel = quietlySound.play();

// associate that sound channel with a SoundTransform.
var quietlyTransform:SoundTransform = new SoundTransform();

// update things like the volume with that sound transform.
quietlyTransform.volume = .5;

// apply the sound transform to the channel.
quietlyChannel.soundTransform = quietlyTransform;
```

Utah State

Sound Transform instance

■ Example (if you need to tweak things)

```
var quietlySound:quietlySnd = new quietlySnd();

// associate the sound with a sound channel, tell it to play:
var quietlyChannel:SoundChannel = quietlySound.play();

// associate that sound channel with a SoundTransform.
var quietlyTransform:SoundTransform = new SoundTransform();

// update things like the volume with that sound transform.
quietlyTransform.volume = .5;

// apply the sound transform to the channel.
quietlyChannel.soundTransform = quietlyTransform;
```

Utah State

Supported Video

File type	Extension	PC	MAC
Active Streaming Format	.asf	•	?
Audio Video Interleaved	.avi	•	•
Digital Video	.dv	•	•
Motion Picture Experts Group	.mpg, .mpeg	•	•
Motion Picture Experts Group 4	.mpg4	•	?
QuickTime Video	.mov	•	•
Windows Media	.wmv	•	?

Adapted from CS4 help file

-Direct X 9.0
-Quicktime 7 (mac), Quicktime 6.5 (pc)

Utah State

Video

- Live stream
- Flash Communication Server
- .FLV/.F4V (external/internal)
- Factors that affect video size
  - Frames per second
  - Picture quality
  - Dimensions (height/width)
  - Compression Algorithm

Video compression in Flash

File	Compression	Use
.FLV	Sorenson Spark	Slower end user machines
	On2 VP6	Faster end user machines (results in smaller files sizes)
.F4V	H.264	Spectrum, "HD", Flash 9 required

Video and Sound

- FLV component plays and persists
- Sound Channels play until stopped
- Work around
  - flash.media.SoundMixer.stopAll();
- Timeline controlled sounds are bad for you ("ghost sound"/bug).