HANDLING IMAGES

DISPLAYING IMAGES

- Loading images in and making textures can take time
- Don't wait until you want to present the images to load them in
- Make your textures at the beginning of your script, then present them on time

HANDLING MOVIE FILES

MOVIES

http://www.gstreamer.com

MOVIES

- 1. OpenMovie to open the movie file
- 2. PlayMovie to start playing
- 3. Loop:
  - GetMovieImage to create frame texture
  - Draw texture and flip screen
- 4. PlayMovie to stop playing
- 5. CloseMovie to close movie file

MOVIES

```
[ moviePtr [duration] [fps] [width] [height] [count
[aspectRatio]]=Screen('OpenMovie', windowPtr, moviefile [, async=0]
[, preloadSecs=1] [, specialFlags1=0][, pixelFormat=4]
[, maxNumberThreads=-1]);
```

MOVIES

```
Screen('PlayMovie', moviePtr, rate, [loop], [soundvolume]);
```

0 = stop playback
1 = play, normal speed
-1 = play, normal speed backwards

0 = mute
1 = max volume

MOVIES

```
[ texturePtr [timeindex]]=Screen('GetMovieImage',
windowPtr, moviePtr, [waitForImage=1], [fortimeindex],
[specialFlags = 0] [, specialFlags2 = 0]);
```

```
function playMovie()

     toTime = inf;
     soundvolume = 1;
     pMovie = [pwd '\Batman.mp4'];

     Screen('Preference', 'SkipSyncTests', 1);

     [wPtr, rect] = Screen('OpenWindow', max(Screen('screens')), 0);

     [movie, dur, fps, width, height] = Screen('OpenMovie', wPtr, pMovie);

     Screen('PlayMovie', movie, 1, 0, soundvolume);

     t=GetSecs();

     while t < toTime
          tex = Screen('GetMovieImage', wPtr, movie);

          if tex <=0
               break;
          end

          Screen('DrawTexture', wPtr, tex);
          t = Screen('Flip', wPtr);

          Screen('Close', tex);
     end

     Screen('PlayMovie', movie, 0);
     Screen('CloseMovie', movie);
     sca;
end
```

HANDLING SOUND

SOUND

- PsychPortAudio is the sound driver for Psychtoolbox
- Type PsychPortAudio in the command window to see all of the subcommands, just like Screen
- Get help on a subcommand just like Screen:
  PsychPortAudio Start?
- Test your audio setup with BasicSoundOutputDemo

SOUND DATA

- Sound data should be in the form of a matrix where each row is one sound channel
- Samples in the vector should range from -1 to 1, where 0 is silent.
- You can create a sound by generating data for a matrix on your own, or you can read in from a wav file

READING FROM WAV FILES

```
Y = wavread(FILE)
[ Y, freq ] = wavread(FILE)
```

READING FROM .AU FILES

```
Y = auread(AUFILE)
[Y, freq ] = auread(AUFILE)
```

READING FROM AUDIOFILES

New Matlab command available in versions 2012b and later, will read many audio formats including WAV, FLAC, MP3, MPEG-4, OGG

```
[Y, freq ] = audioread()
```

STEPS TO PLAYING A SOUND

- InitializePsychSound
- open audio channel with PsychPortAudio('Open')
- fill audio buffer with PsychPortAudio('FillBuffer')
- start playing a sound with PsychPortAudio('Start')
- stop playing a sound with PsychPortAurio('Stop')
- close the audio channel with PsychPortAudio('Close')

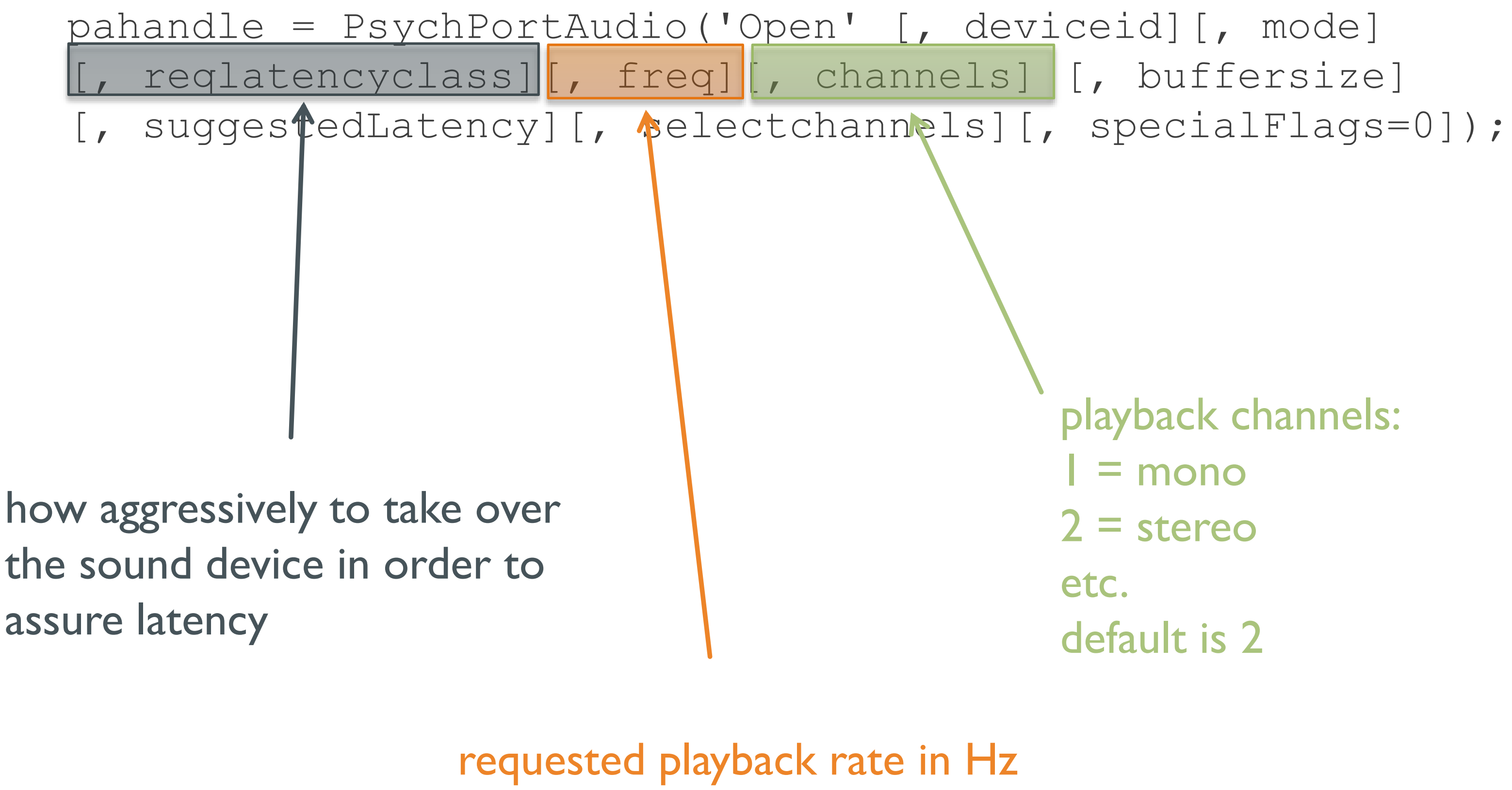
WHY DO WE USE PSYCHPORTAUDIO FUNCTIONS?

- We have learned about 'audioplayer.m' function that can play any audio files in matlab. Why do we have to use PsychoPortAudio function?

STEP 1: INTIALIZE

- InitializePsychSound
  - Loads the sound driver. Place this at the beginning of your script.
  - on Windows, things may not work with high precision timing without an ASIO sou nd card (read help InitializePsychSound if you are on Windows)

STEP 2: OPEN AUDIO CHANNEL



```
pahandle = PsychPortAudio('Open' [, deviceid][, mode]
[, reqlatencyclass][, freq][, channels] [, buffersize]
[, suggestedLatency][, selectchannels][, specialFlags=0]);
```

how aggressively to take over the sound device in order to assure latency

requested playback rate in Hz

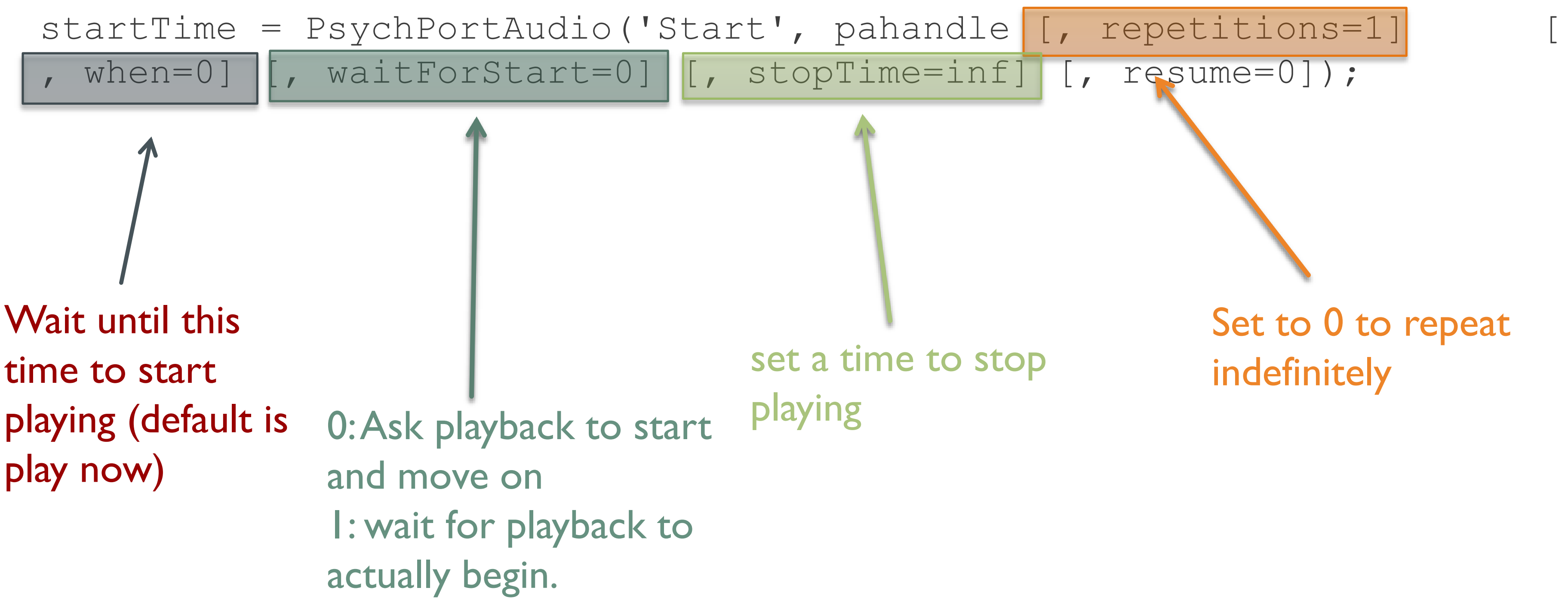
playback channels:
1 = mono
2 = stereo
etc.
default is 2

STEP 3: FILL THE AUDIO BUFFER

```
PsychPortAudio('FillBuffer', pahandle, bufferdata);
```

This is analogous to drawing on the back buffer with the Screen command. We fill the buffer now, but it will not be heard until we play it.

STEP 4: START PLAYBACK



```
startTime = PsychPortAudio('Start', pahandle [, repetitions=1]        [
, when=0] [, waitForStart=0] [, stopTime=inf] [, resume=0]);
```

Wait until this time to start playing (default is play now)

0: Ask playback to start and move on
1: wait for playback to actually begin.

set a time to stop playing

Set to 0 to repeat indefinitely

REMAINING STEPS

- Stop playback if necessary: PsychPortAudio('Stop',pahandle);
- Close the audio driver: PsychPortAudio('Close',pahandle);

```
function playSound()
    scriptStart = GetSecs();
    InitializePsychSound;
    wavefilename = [PsychtoolboxRoot 'PsychDemos' filesep 'SoundFiles' filesep 'funk.wav'];
    [soundData, freq] = audioread(wavefilename);

    soundData = transpose(soundData);
    soundData = [soundData;soundData];
    numChannels = 2;

    soundLength = size(soundData,2);
    soundDur = soundLength/freq;

    pahandle = PsychPortAudio('Open', [], [], 0, freq, numChannels);
    PsychPortAudio('FillBuffer', pahandle, soundData);
    stTime = GetSecs();
    playTime = PsychPortAudio('Start', pahandle, [], [], 1);
    fprintf('Sound started playing %.2f seconds after start of script\n', stTime+playTime-
scriptStart);

    WaitSecs(soundDur);

    PsychPortAudio('Stop', pahandle);
    PsychPortAudio('Close', pahandle);
end
```

SOUND RECORDING STEPS

- Initialize sound driver: InitializePsychAudio
- Open audio channel for recording with PsychPortAudio('Open') setting mode to 2
- Clear a buffer using PsychPortAudio('GetAudioData')
- Start recording with PsychPortAudio('Start')
- Stop recording with PsychPortAudio('Stop')
- Get audio data using PsychPortAudio('GetAudioData')

STEP 2: OPEN AUDIO CHANNEL

```
pahandle = PsychPortAudio('Open' [, deviceid][, mode]
[, reqlatencyclass][, freq][, channels] [, buffersize]
[, suggestedLatency][, selectchannels][, specialFlags=0]);
```

1: sound playback only (default)
2: audio capture
3: simultaneous capture and playback (may not work on all hardware)

GETAUDIODATA

Call before you start recording to setup an empty buffer, then after recording to retrieve recorded data

```
[audiodata absrecposition overflow cstarttime] = PsychPortAudio('GetAudioData',
pahandle [, amountToAllocateSecs]  [, minimumAmountToReturnSecs][, maximumAmount
ToReturnSecs]         [, singleType=0]);
```

WRITING DATA TO FILE

- wavwrite(audiodata, freq, nbits, filename)
- audiowrite(filename, audiodata, freq)

```
function audioData = recordSound

    InitializePsychSound;
    duration = 5;

    freq = 44100;
    pahandle = PsychPortAudio('Open', [], 2, 0, freq, 2);

    PsychPortAudio('GetAudioData', pahandle, duration);

    PsychPortAudio('Start', pahandle, 0, 0, 1);

    fprintf('Recording...\n');
    WaitSecs(duration);
    fprintf('Done recording.\n');

    PsychPortAudio('Stop', pahandle);

    audioData = PsychPortAudio('GetAudioData', pahandle);

    PsychPortAudio('Close', pahandle);

end
```