Core Audio &
The Amazing Audio Engine

iOS KW – September 2015

Code & slides are available at http://github.com/jayrhynas/Audio-iOSKW

Jayson Rhynas · @jayrhynas

MEDLY

WHO IS THIS JAYSON GUY?

???

WHO IS THIS JAYSON GUY?

CORE AUDIO

Set of frameworks for working with audio

Varying levels of abstractions

Includes utilities for loading, saving, and converting between audio file formats

Also includes classes for playing sound effects & audio files

CORE AUDIO

OpenAL: cross-platform API for positioning and playing sounds

- analogous to OpenGL (shares same coordinate system)
- good for games

Audio Units: lowest level API (on iOS)

- all other Core Audio frameworks are built on top

AUDIO UNITS

Self-contained plugins that process audio

Prior to iOS 9, could only use built-in audio units provided by Apple

- iOS 9 introduced support for 3rd party plugins (Audio Unit Extensions)

Apple provided sample of Audio Unit Extensions:

https://developer.apple.com/library/ios/samplecode/AudioUnitV3Example/Introduction/Intro.html

AUDIO UNITS

Audio units either generate, process, or consume audio

AUSampler is a sample-based instrument that outputs audio

AULowpassFilter is an effect that takes input audio, transforms it, and outputs it

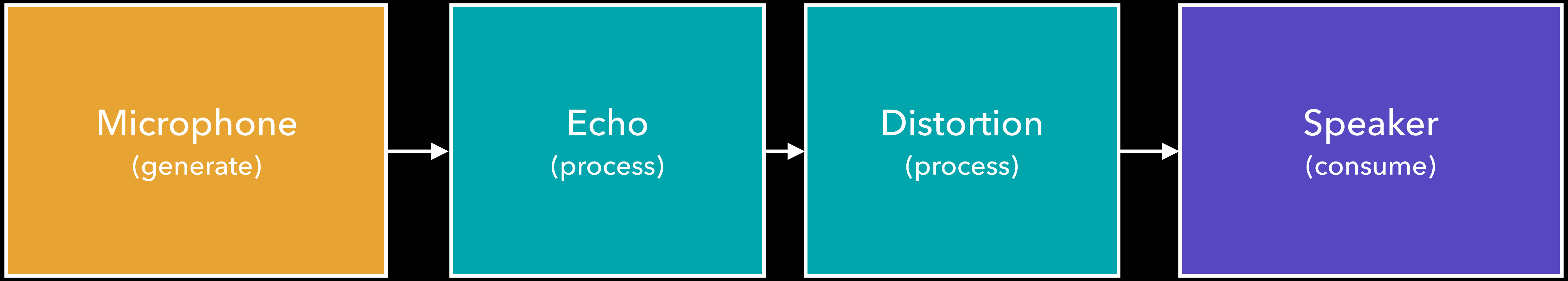
AURemoteIO interacts with the device hardware

- audio fed into the input is sent to the speakers
- audio from the microphone is sent to the output

AUDIO UNITS

You connect the output of one audio unit to the input of another, forming a processing chain

Remember: Audio units either generate, process, or consume audio



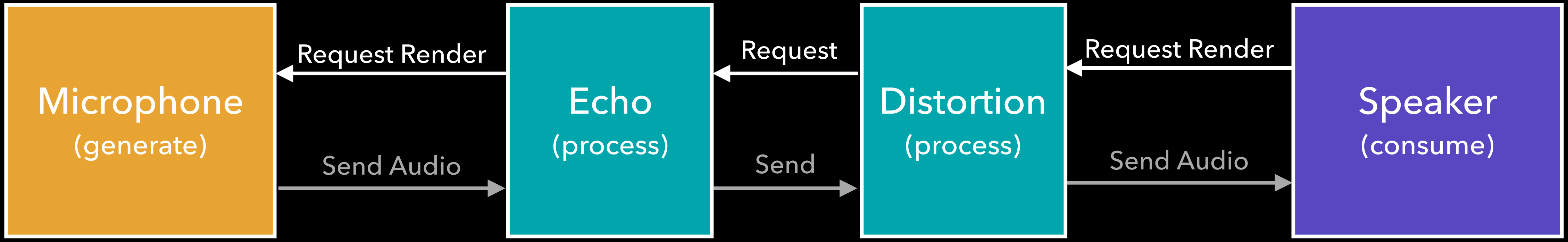
AUDIO UNITS

Audio units use a pull model:

The IO unit asks for some audio from the unit connected to its input, which in turn asks for audio from its upstream units, processes the audio, and returns it to the downstream unit

Can also attach callbacks (C functions) to inputs

- callback will be passed a buffer that is expected to be filled with audio
- used for custom audio generation & filtering



AUDIO GRAPH

Provides an abstraction for adding & removing audio unit nodes in a processing graph

Allows safe updates to processing chain while audio is playing

CORE AUDIO

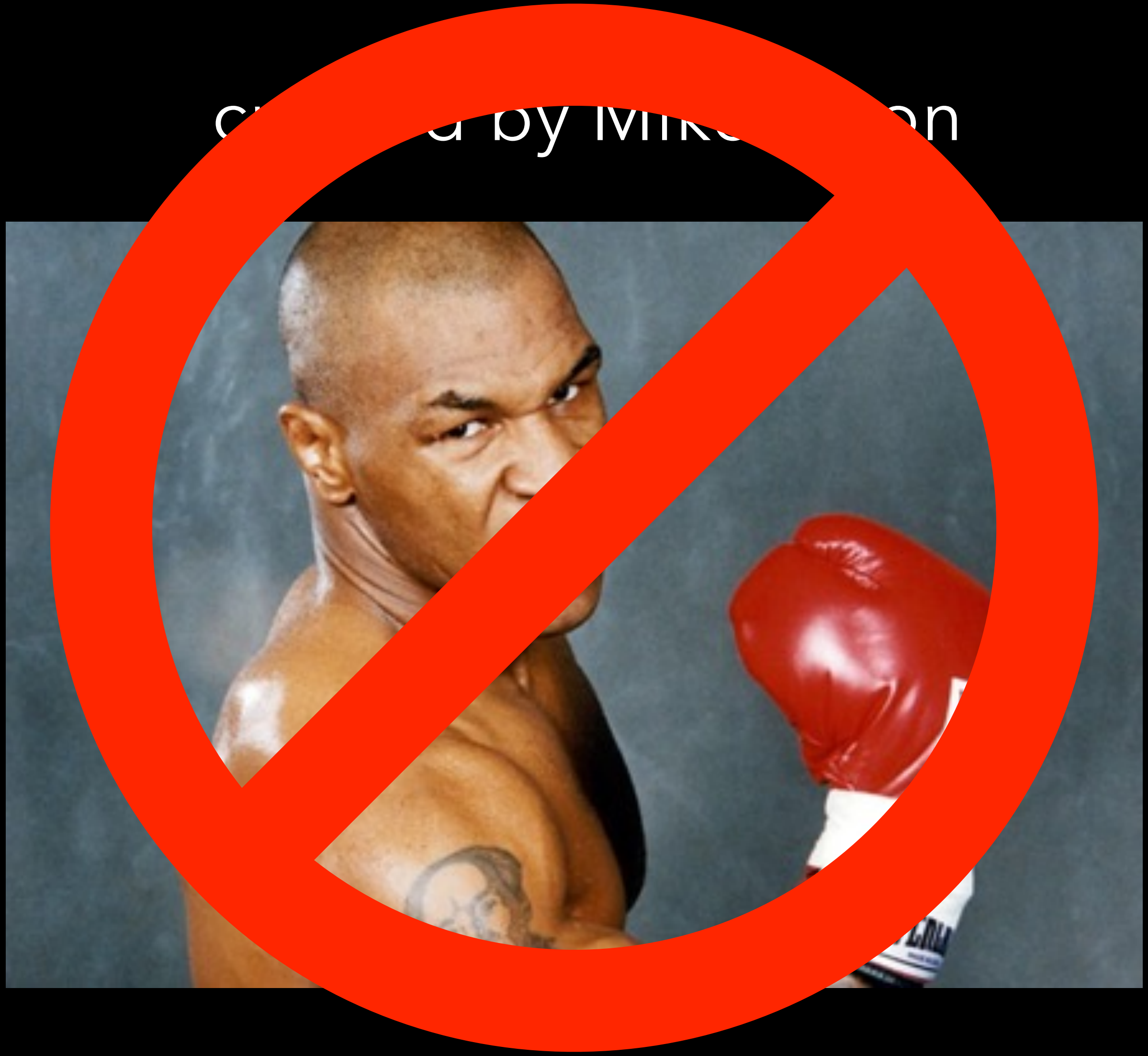
Audio Unit and Audio Graph are C-based APIs

Steep learning curve, can be difficult to properly configure and manage a processing chain

- have to make sure audio formats match, threading considerations

THE AMAZING AUDIO ENGINE

created by Mike Tyson



created by Michael Tyson

Creator of Loopy,
AudioBus apps

THE AMAZING AUDIO ENGINE

Provides a set of convenient Objective-C classes for managing iOS audio

- uses Audio Units & Audio Graph underneath

Still lets you use audio units & your own processing methods where necessary

Demo

Code & slides are available at
http://github.com/jayrhynas/Audio-iOSKW

Email
jayson@medlylabs.com