Intelligent implementation

Keeping the Program Lead off your back

Alistair Hirst
General Manager

OMNI Audio

GAME DEVELOPERS CONFERENCE 2012
SAN FRANCISCO, CA
MARCH 5-9, 2012
EXPO DATES: MARCH 7-9

Introduction

- Started in games 1990
  - Sega Genesis:
    - 6 voices of FM
    - 5 voices and one voice 8 bit 11kHz sample

Introduction

- In 20 years
  - VR Worlds, jacked in
  - still fighting with simulation and graphics for resources.

The Problem

- Resources are limited

The Problem

- Resources are limited
- Trade offs need to be made
  - Biggest bang for the buck

The Problem

- Resources are limited
- Trade offs need to be made
  - Biggest bang for the buck
- Trade offs are different depending on the platform and game. Find the bottleneck and balance.

Trade offs

Number Voices

CPU Cycles

Fidelity

Streams

Effect Quality

Sample Rate

RAM

Latency

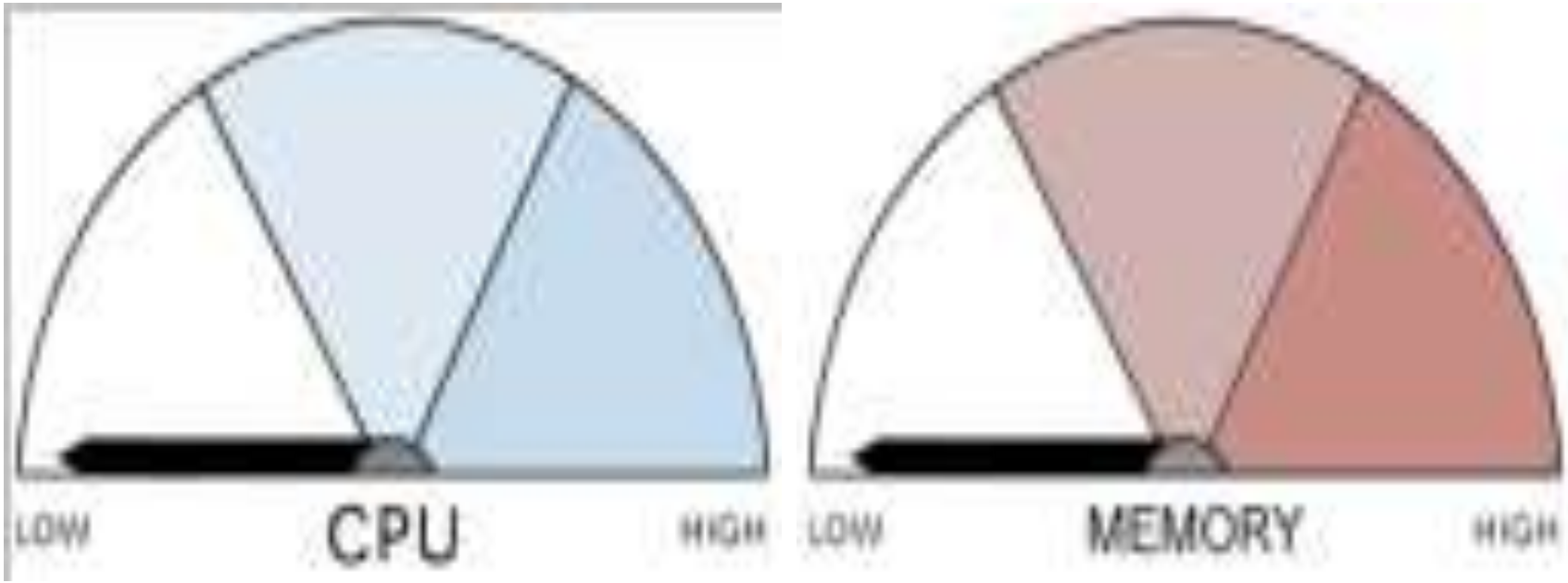
Game Event Driven Changes

Data Compression

Buffer size

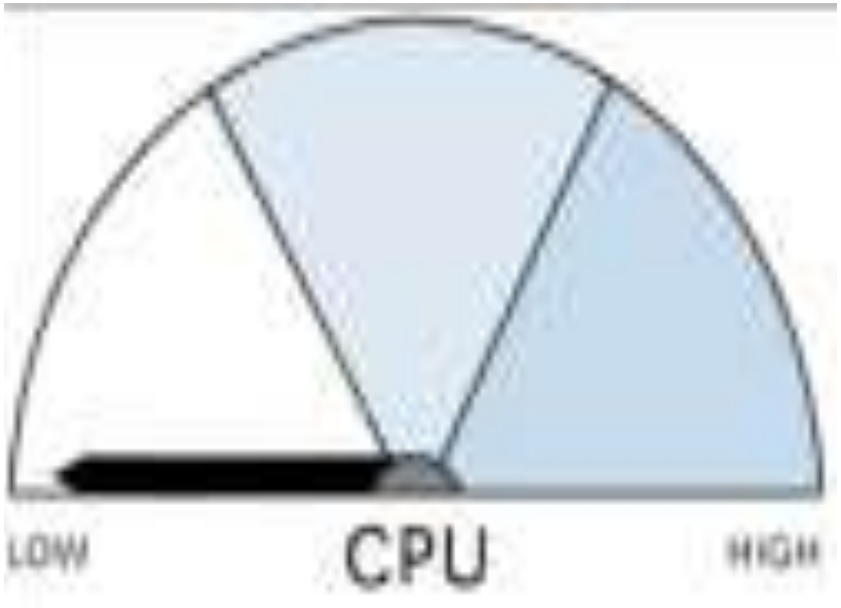
FMOD CPU/Memory hits:

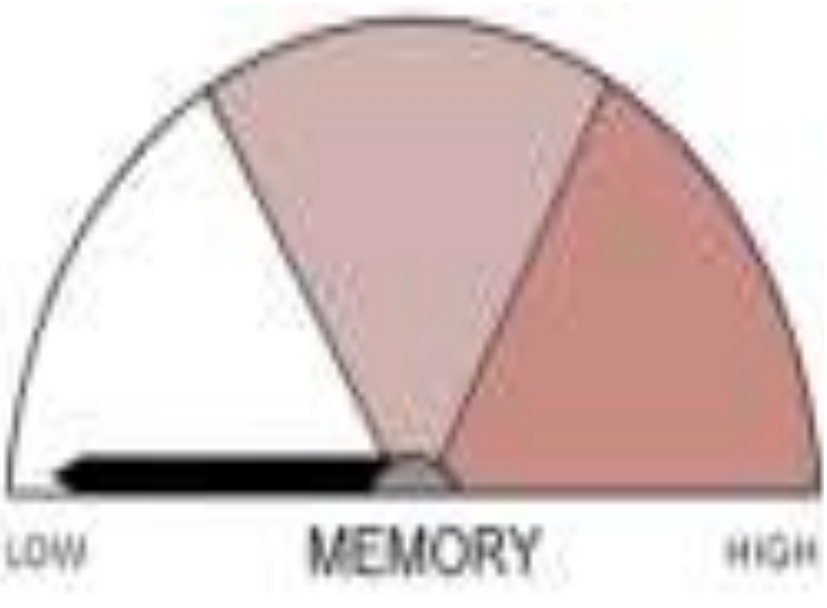
Volume



FMOD CPU/Memory hits:

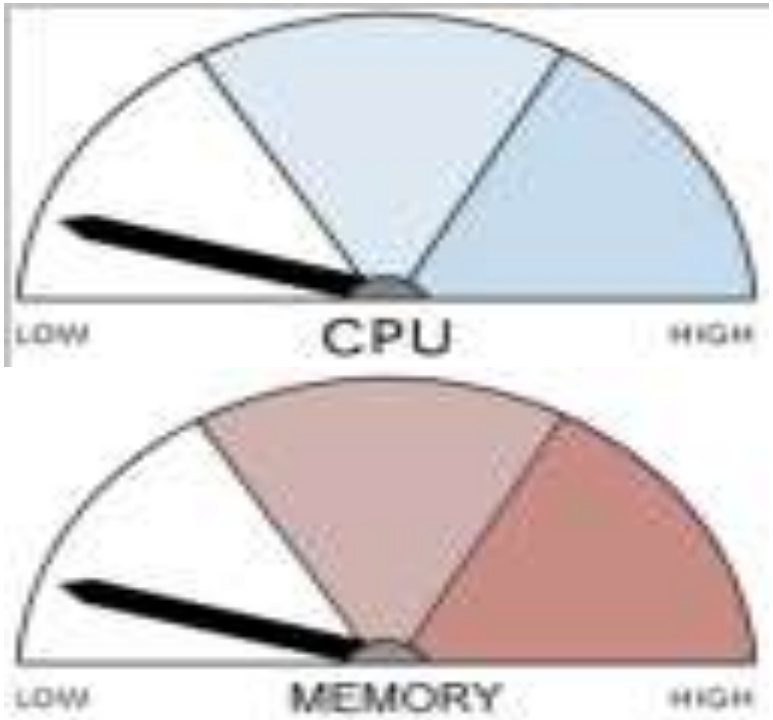
Surround Pan





FMOD CPU/Memory hits:

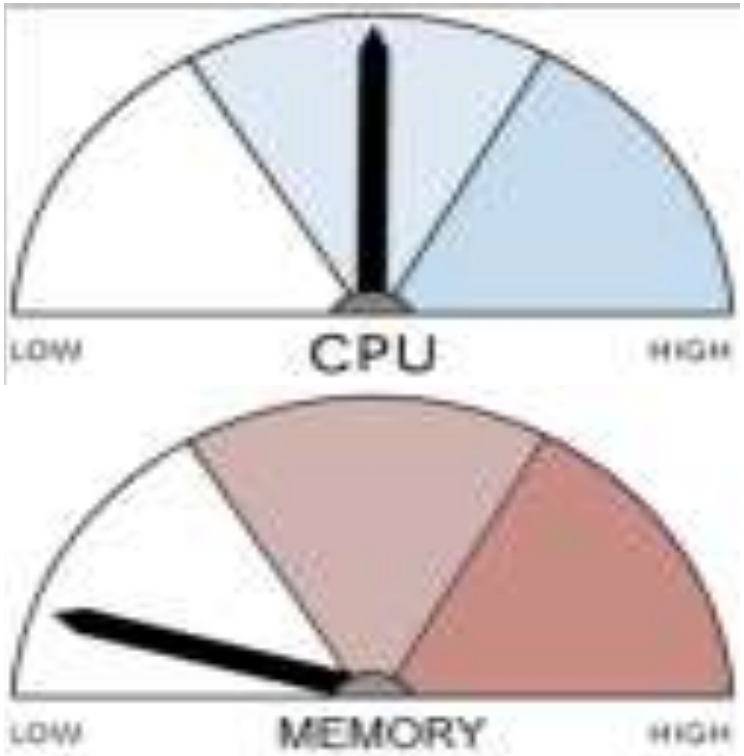
LoPass Simple



FMOD CPU/Memory hits:

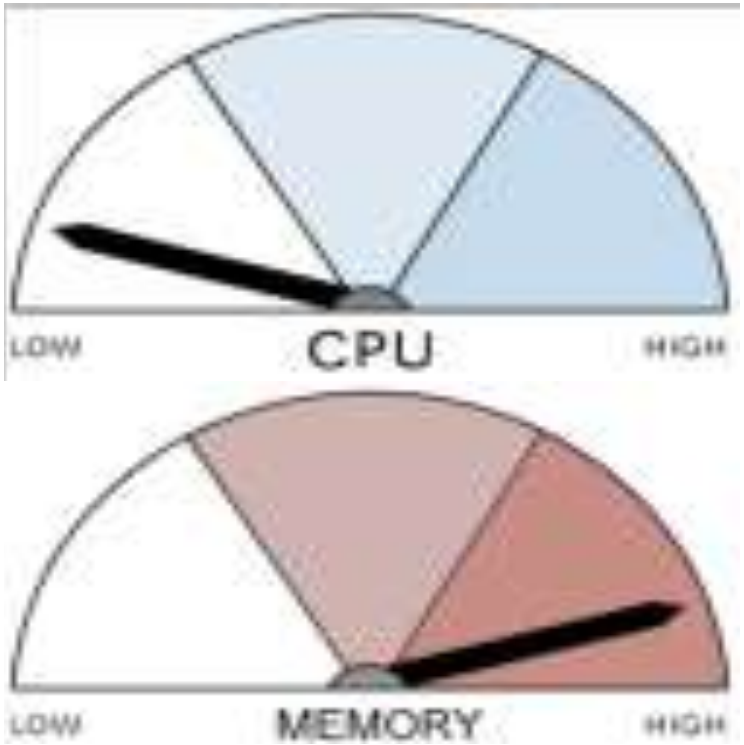
FMOD Lopass filter

(analog resonance emulation)



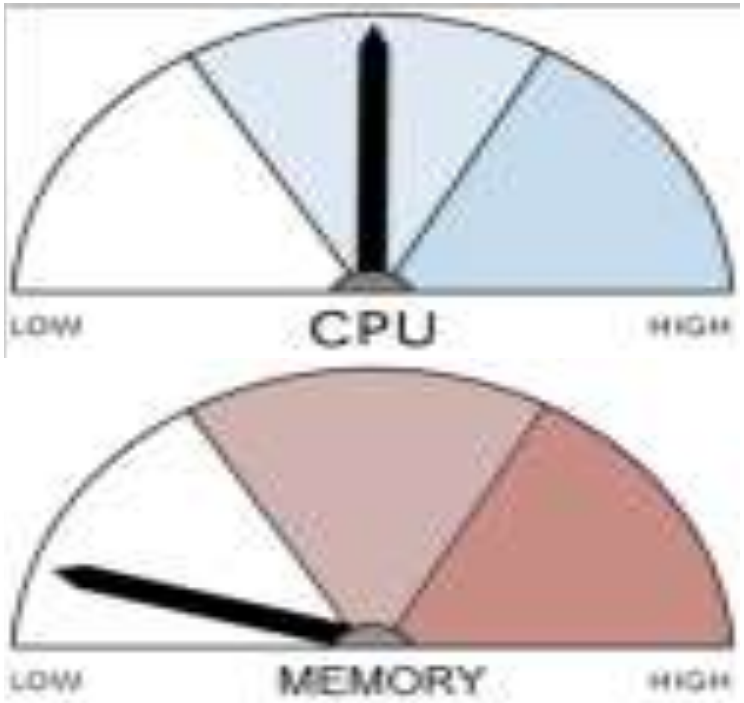
FMOD CPU/Memory hits:

Echo



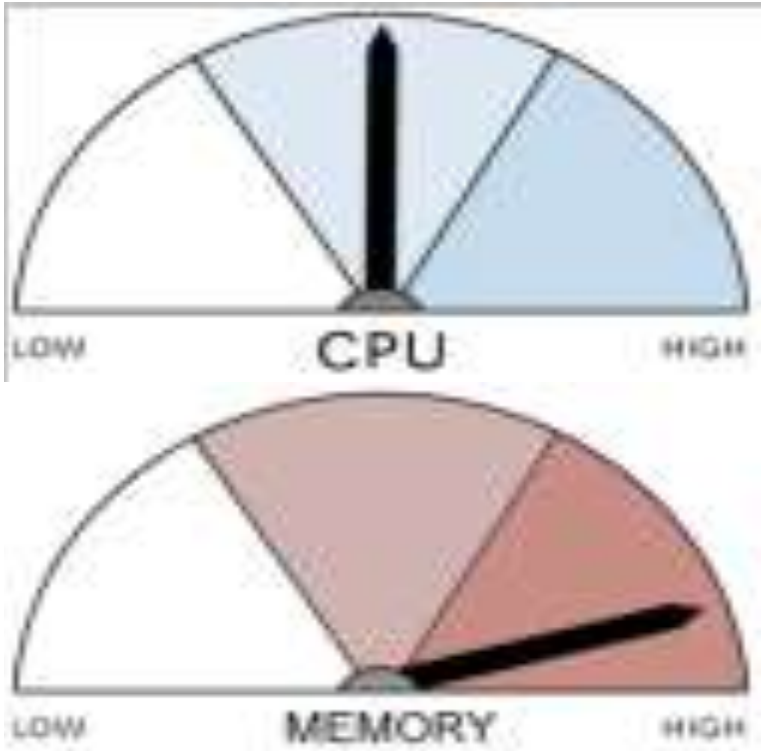
FMOD CPU/Memory hits:

Parametric EQ



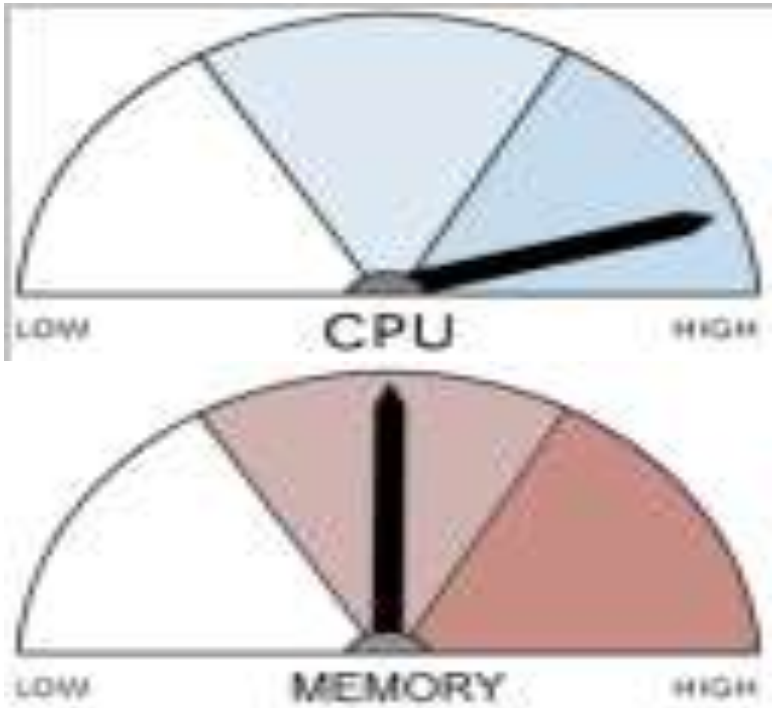
FMOD CPU/Memory hits:

Chorus



FMOD CPU/Memory hits:

FMOD SFX Reverb



Trade offs

- CPU cycles
- RAM
- Number of voices
- Hard Drive streams
- Optical Disk streams
- Latency/buffer sizes
- Sample rate/data compression/fidelity
- Game event driven changes
- Effect quality/CPU hit

Trade offs

- CPU Cycles vs RAM
  - Decompressing mp3 on the fly (runtime)

    Versus
  - Decompressing mp3 into RAM

Trade offs

- # Streams vs buffer size vs latency
  - The read head needs time to seek

Trade offs

- # Streams vs buffer size vs latency
  - The read head needs time to seek
  - More streams -> Bigger buffers

Trade offs

- # Streams vs buffer size vs latency
  - The read head needs time to seek
  - More streams -> Bigger buffers
  - Bigger buffers -> higher latency

Trade offs

- Number of Voices Vs CPU
  - Each voice uses CPU cycles

Trade offs

- Number of Voices Vs CPU
  - Each voice uses CPU cycles
  - More voices allow layering, more immersion

Trade offs

- Effect quality vs CPU hit
  - Higher quality EQ or reverb -> more CPU

Case Studies: iPhone

- Bottleneck: Download under 20MB to go over 3G network – footprint potentially limited.

Case Studies: iPhone

- Music compressed as AAC to make smaller
- iPhone can only decompress 1 AAC/mp3 stream at a time in hardware

Case Studies: iPhone

- Result: SFX shouldn’t be AAC, will affect frame rate.
  - Reduce sample rate to reduce size

Case Studies: iPhone

- Result: SFX shouldn't be AAC, will affect frame rate.
  - Reduce sample rate to reduce size.
  - Or use IMA 4:1 compression

Case Study: PC downloadable

- Bottleneck: Size needs to be minimized
  - to reduce bandwidth costs and
  - Reduce download time

Case Study: PC downloadable

- Data reduction on audio assets
  - mp3,
  - Ogg Vorbis

Case Study: PC downloadable

- Decoding MP3's at run time causes CPU hit
  - Possible frame rate drop, pause

Case Study: PC downloadable

- Decoding MP3's at run time causes CPU hit
  - Possible frame rate drop, pause
- PC's have more RAM than consoles

Case Study: PC downloadable

- Solutions:
  - decompresss into RAM before playing
    - (pre-cache in Flash)

Case Study: PC downloadable

- Solutions:
  - decompresss into RAM before playing
    - (pre-cache in Flash)
  - use IMA 4:1 compression. Lighter CPU, less size reduction.

Case study: Xbox 360

- Bottleneck:
  - RAM

Case study: Xbox 360

- Bottleneck:
  - RAM
  - potentially DVD space,

Case study: Xbox 360

- Bottleneck:
  - RAM
  - potentially DVD space
  - limited # of streams off of DVD

Case study: Xbox 360

- XMA compressed files decoded on hardware
  - no CPU hit
  - Lots of voices

Case study: Xbox 360

- Solution:
  - XMA Compression

Case study: Xbox 360

- Solution:
  - XMA Compression
  - Tradeoff compression/fidelity needed

Case study: Xbox 360

- Solution:
  - XMA Compression
  - Tradeoff compression/fidelity needed
  - Minimize streaming

Game audio middleware != DAW

- Game effects are optimized for CPU efficiency

Game audio middleware != DAW

- DAW's have more resources dedicated to audio

Game audio middleware != DAW

- DAW's have more resources dedicated to audio
- DSP cards like UAD can have even higher quality emulations because of dedicated DSP

Game audio middleware != DAW

- DAW's have more resources dedicated to audio
- DSP cards like UAD can have even higher quality emulations because of dedicated DSP
- External hardware available with DAW

Game audio middleware != DAW

- Better sound quality

Game audio middleware != DAW

- Better sound quality
- Much more variety in models for compressors, reverb, sweeter EQ’s.

General Rule

- If it’s not changing in realtime in-game, consider baking the effect in

Common Mistakes

- Applying EQ or filters that never change to sounds

Common Mistakes

- Applying EQ or filters that never change to sounds
- Applying compression to a single sound source that never changes

Common Mistakes

- Applying distortion that never changes to a sound source to make it more audible

Common Mistakes

- Applying distortion that never changes to a sound source to make it more audible
- Using multiple layers of sounds non-dynamically

Realtime changes are good

- increases immersion
- provides realtime feedback to player

Realtime changes are good

- Choose relevant game parameters to drive them:
  - Distance

Realtime changes are good

- Choose relevant game parameters to drive them:
  - Distance
  - Force

Realtime changes are good

- Choose relevant game parameters to drive them:
  - Distance
  - Force
  - Speed
  - Location

Realtime effects:

- dynamic EQ/filters fed by RTPC tied to game states
  - Filter or EQ on damage sounds, tied to force

Realtime effects:

- compression or limiting on the master bus
  - Can only be done realtime, keeps mix under control.

Realtime effects:

- Dynamic distortion
  - Trash distortion used in Forza 4, driven by load on engine.

Realtime effects:

- Dynamic distortion
  - Trash distortion used in Forza 4, driven by load on engine.
  - CPU load (of 1 thread, ½ core)
    - Trash = 3 - 5 %
    - FMOD = 1%

Realtime effects:

- Chorus or flanging effects best done at runtime
  - realtime chorus will help mask loop points, sounds more natural

Streaming

- HD seeks much faster than DVD

Streaming

- HD seeks much faster than DVD
- Switching layers on DVD is slow

Streaming

- HD seeks much faster than DVD
- Switching layers on DVD is slow
- Read heads only go so fast
  - Bigger the buffer, the more time available
  - More streams available
  - Higher latency

Streaming

- Probably sharing with other game data
  - Textures
  - Geometry

Streaming

- Speech and music typically streamed
  - Except when it isn’t.

Streaming

- Have to coordinate with lead programmer
  - Allocate resources as early as possible
  - Stream not available? Argue for more RAM

Streaming

- Ambiences:
  - Try overlapping loops
  - Random one-shots

Voice management

- Need to limit number of voices playing
  - Shut off voices out of audible range

Voice management

- Need to limit number of voices playing
  - Shut off voices out of audible range
  - Take up CPU tracking position and volume updates

Voice management

- Offload to “virtual voices”
  - Will play at right position when come back in range

Voice management

- Offload to “virtual voices”
  - Will play at right position when come back in range
  - Music

Voice management

- Offload to “virtual voices”
  - Will play at right position when come back in range
  - Music
  - Retrigger loops

Voice management

- Use a Priority system to cull voices
  - Oldest
  - Quietest
  - Most important to least important

Voice management

- intelligent instance limiting
  - Group sounds into categories, and limit the number that can play at once. (example: bird chirps, wind loops).

Tips & Tricks

- Don’t place individual sound points for ambient sounds (birds, insects)
  - Tracking all those points is CPU intensive

Tips & Tricks

- Don’t place individual sound points for ambient sounds (birds, insects)
  - Tracking all those points is CPU intensive
- designate an area where birds are heard and play with random positioning and volume

Tips & Tricks

- Stream one voice to multiple locations
  - Example: Speakers around a race track
  - Saves on voice overhead

Tips & Tricks

- Categorize your sounds, and put a voice limit on that category to avoid making your mix too dense.

Questions?