

GD10C

Learn. Network. Inspire.

www.GDConf.com

A Bioshock 2 Post-Mortem

Michael Kamper

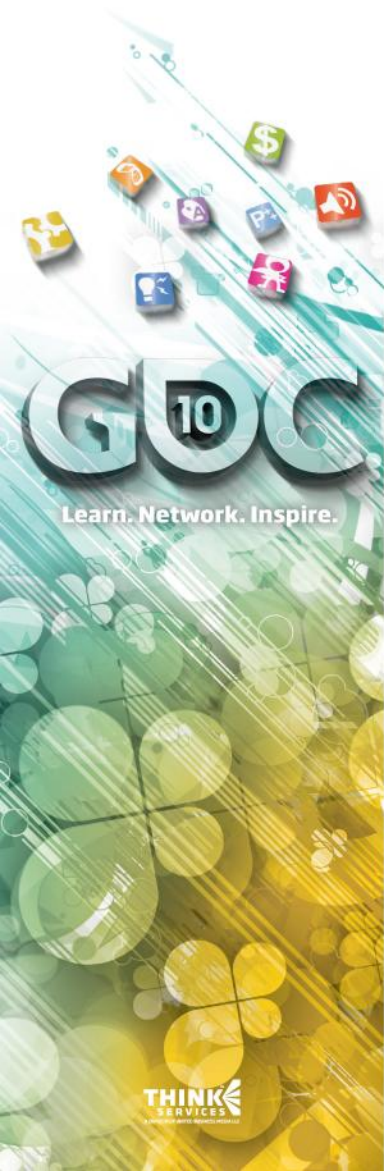
2K Marin Audio Lead

Michael Csurics

2K Marin Dialogue Supervisor

Guy Somberg

2K Marin Audio Programmer



Single Player Audio Staff

David Steinwedel

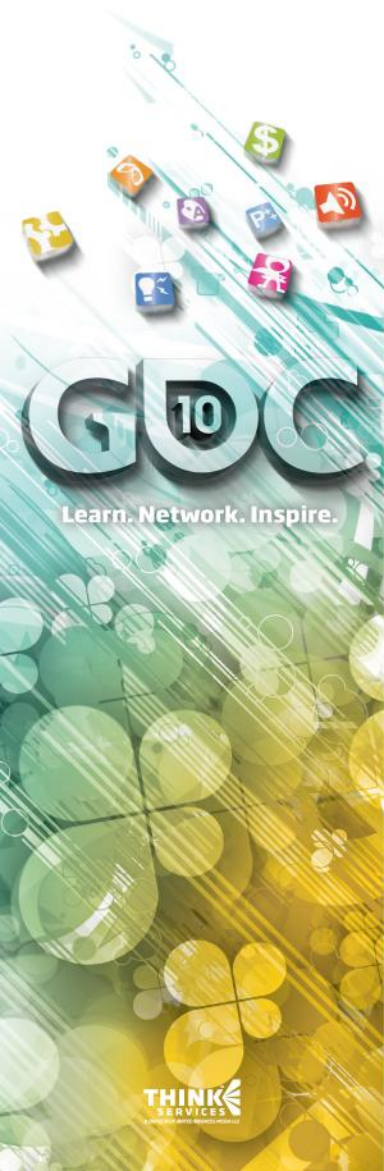
2K Marin Senior Sound Designer

Justin Mullins

2K Australia Sound Design/Implementation

Andy Lackey

2K Marin Contract Sound Design/Implementation



Single Player Additional Audio

David Farmer

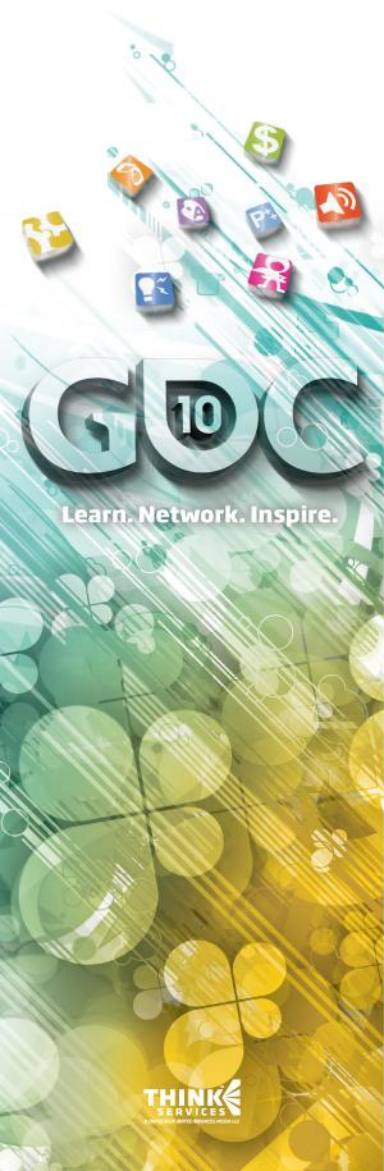
Contract Sound Design

dSonic

Contract Sound Design

One Step Up

Foley Recording



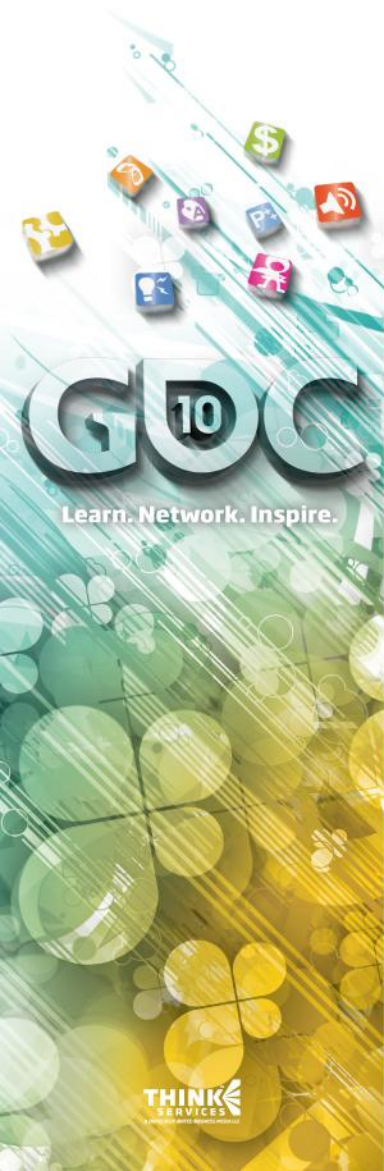
Multi Player Audio Staff

George Spanos

Digital Extremes Audio Lead

Dustin Crenna

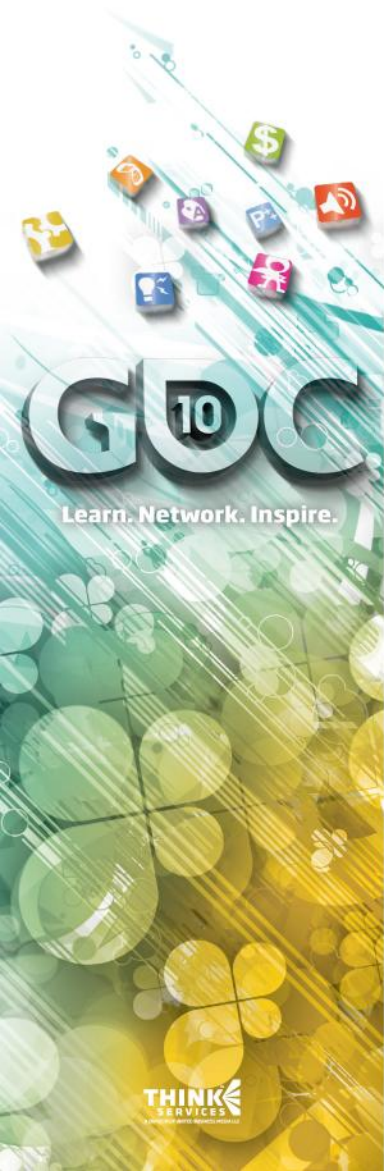
Digital Extremes Audio Designer



Music

Garry Schyman

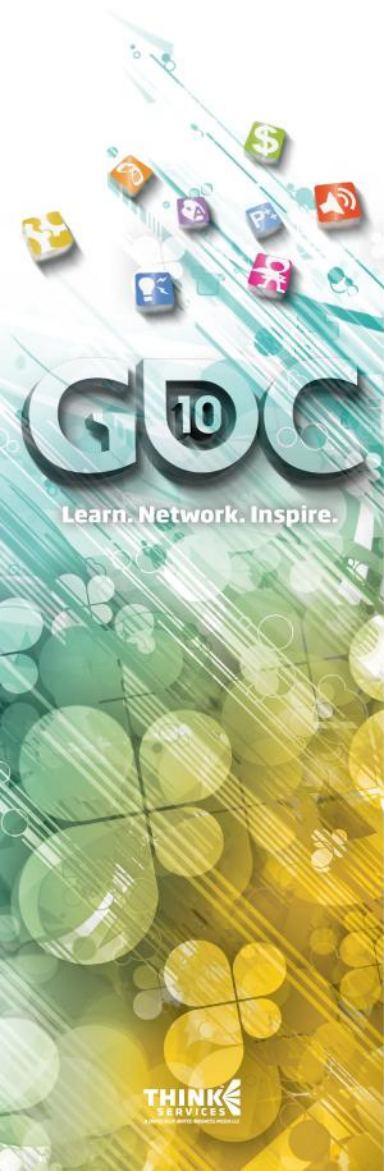
Composer



Challenges

Live Up To A Classic

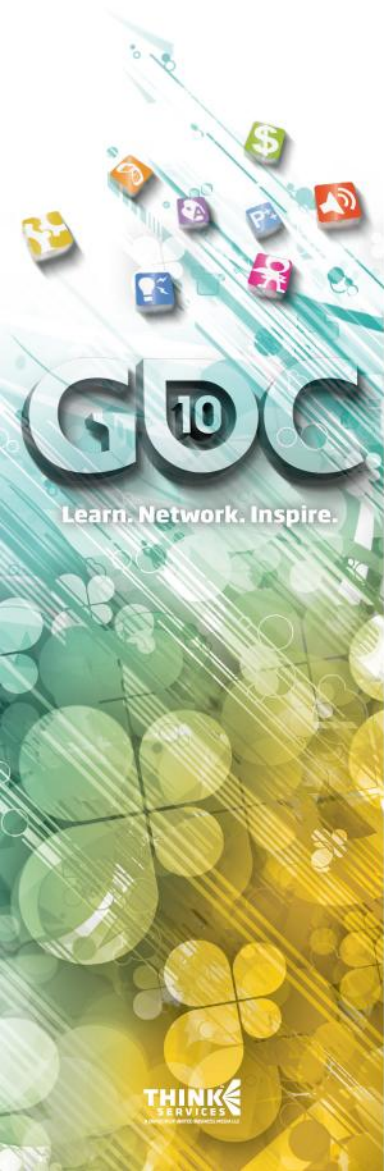
- ⌚ Bioshock universally lauded for audio
- ⌚ Rich environments and characters
- ⌚ Established style and framework
- ⌚ Created shorthand for world elements



Challenges

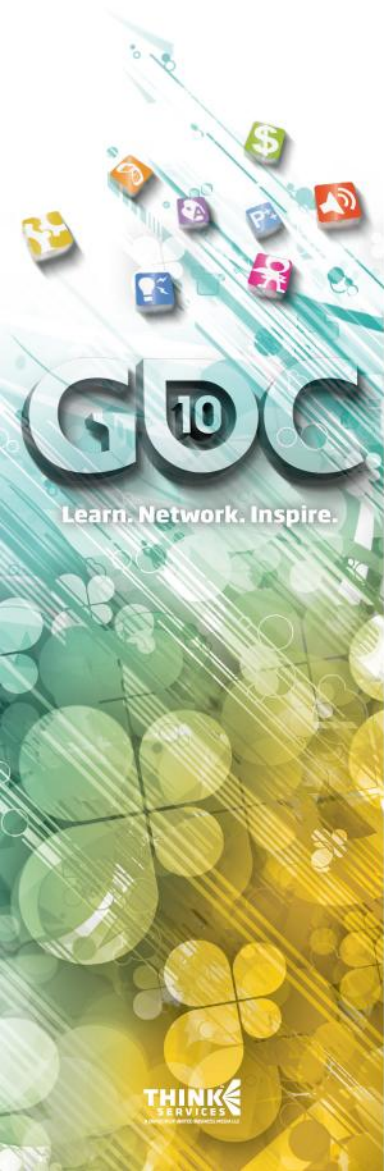
Legacy Audio Engine/Implementation

- ⊗ Completely text-based
- ⊗ Extended baking time
- ⊗ Counter-intuitive



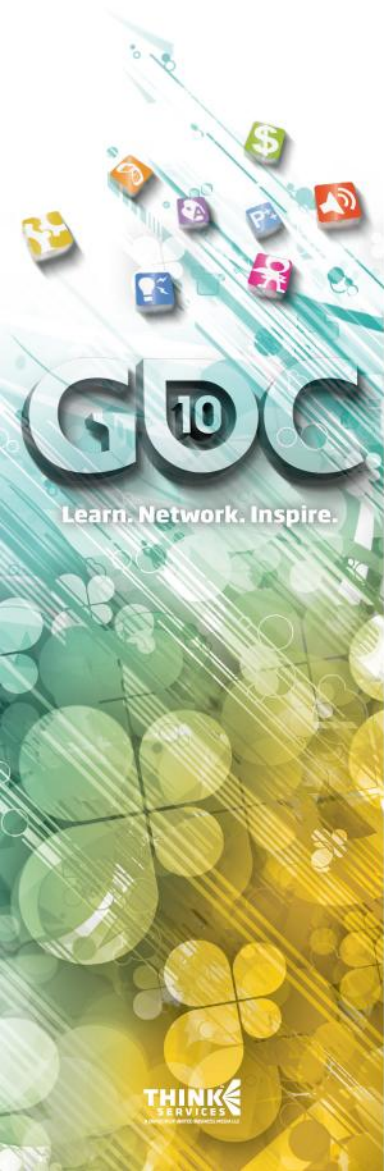
Building the Foundation

- ③ Bioshock 1 engine was showing its age
 - ③ Slow iteration time
 - ③ Complicated integration procedure
 - ③ No modern audio engine features
 - ③ Myriad code problems
- ③ Time for a rewrite
 - ③ Do what you want
 - ③ But don't break it!

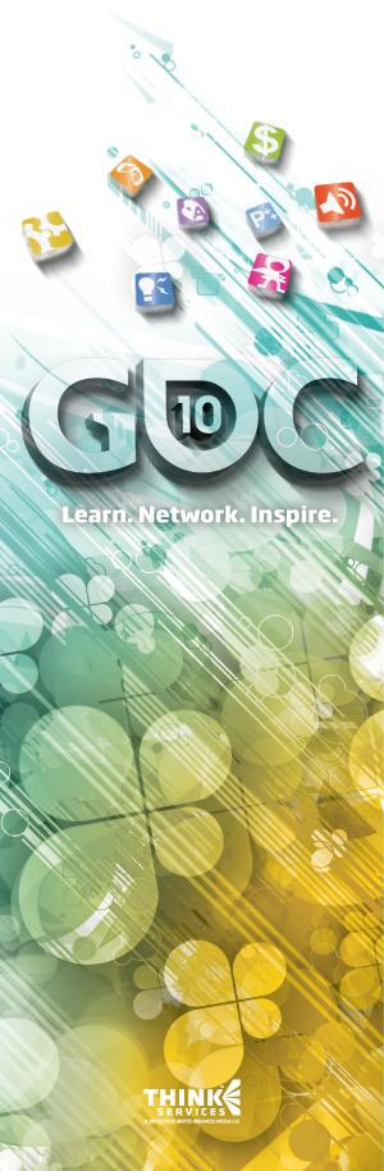
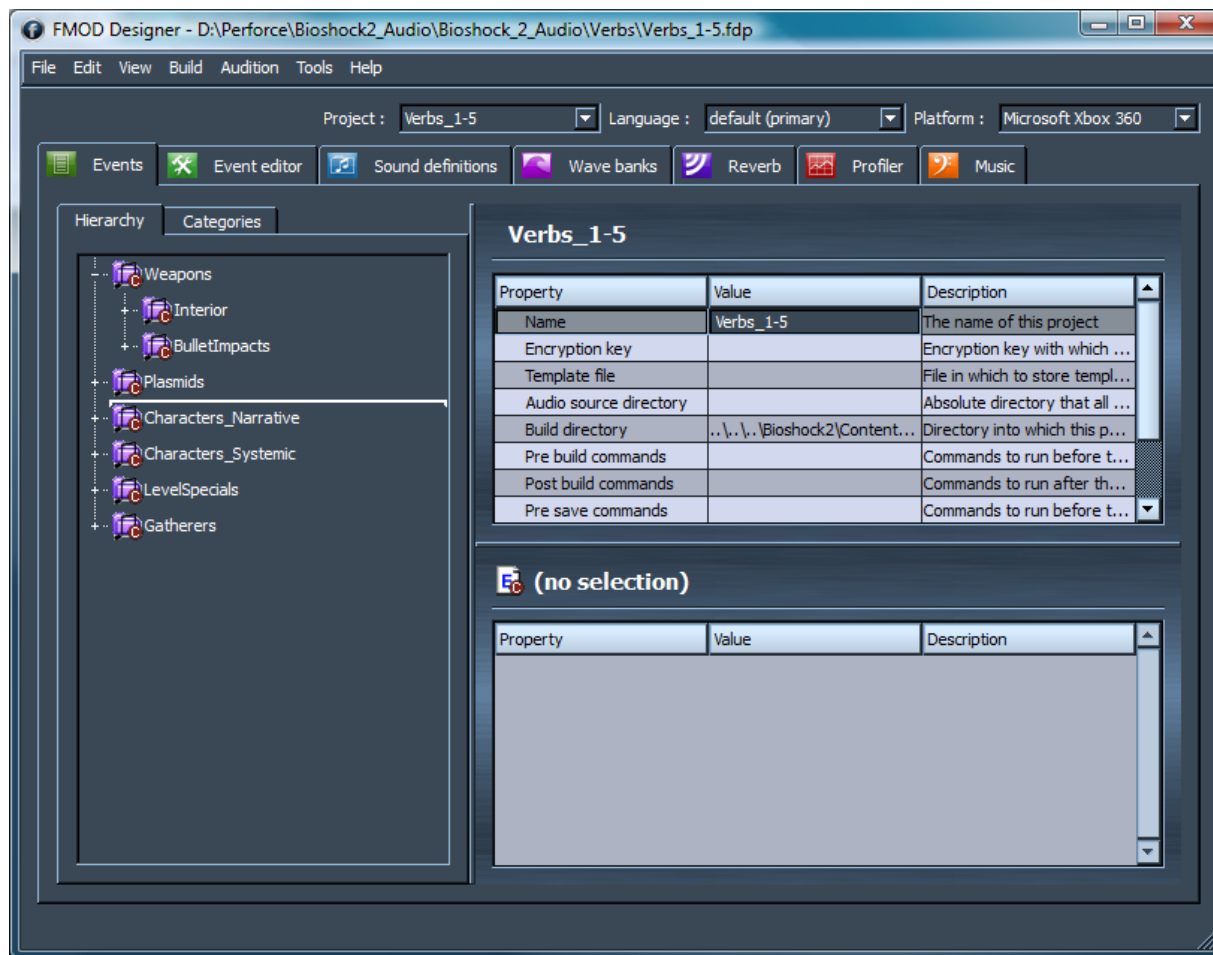


Audio Engine Goals

- ⌚ Fast iteration time
- ⌚ Make sound design and integration easy and flexible
- ⌚ Provide modern audio engine features
- ⌚ But don't break it!
- ⌚ Conclusion:
 - ⌚ Use FMOD Designer Tool



FMOD Designer Tool



Sound Design

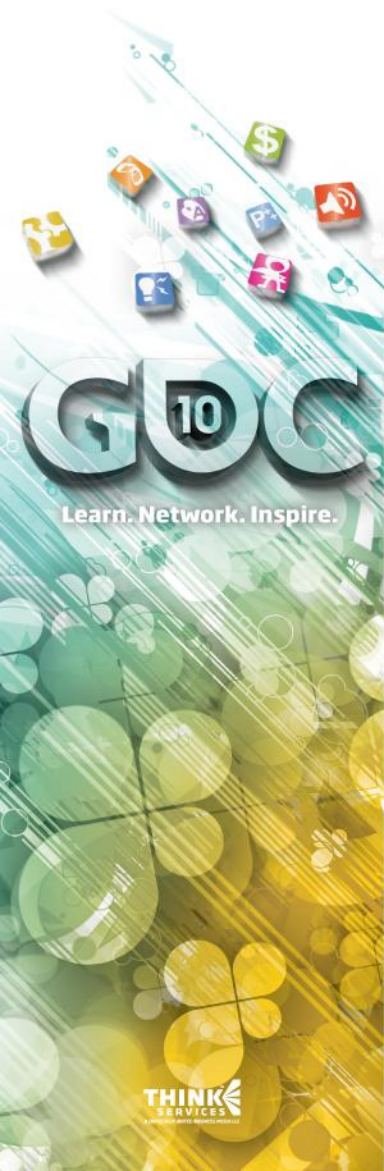
FMOD Design Tool

⌚ Positives

- ⌚ Dynamic Design Capabilities
- ⌚ Fast Iteration and Implementation

⌚ Negatives

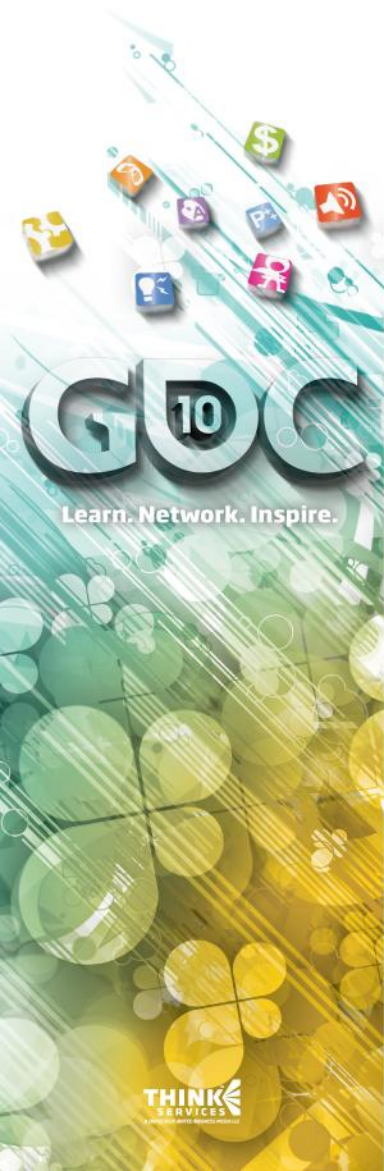
- ⌚ Increased System Memory



Sound Design

Creative Approach

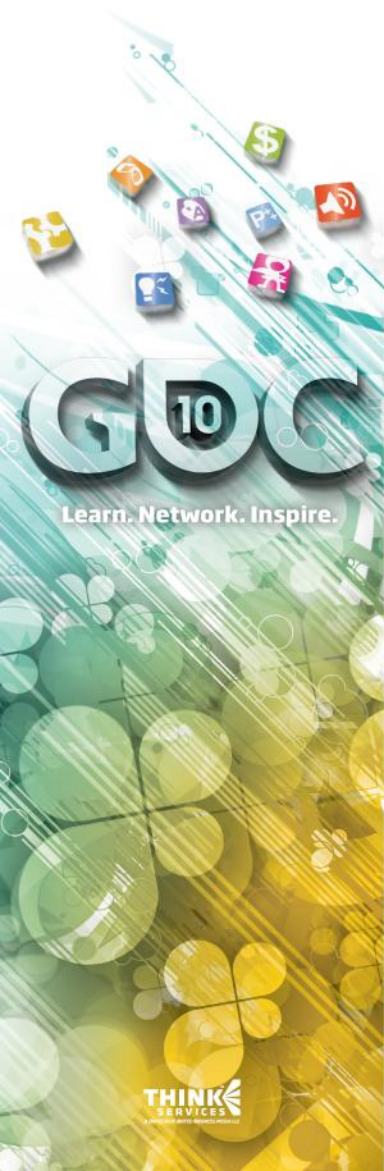
- ⌚ Part of that World
 - ⌚ Living Ambience
 - ⌚ Interactive Ambience
 - ⌚ It's not real if it doesn't make a sound



Sound Design

Creative Approach

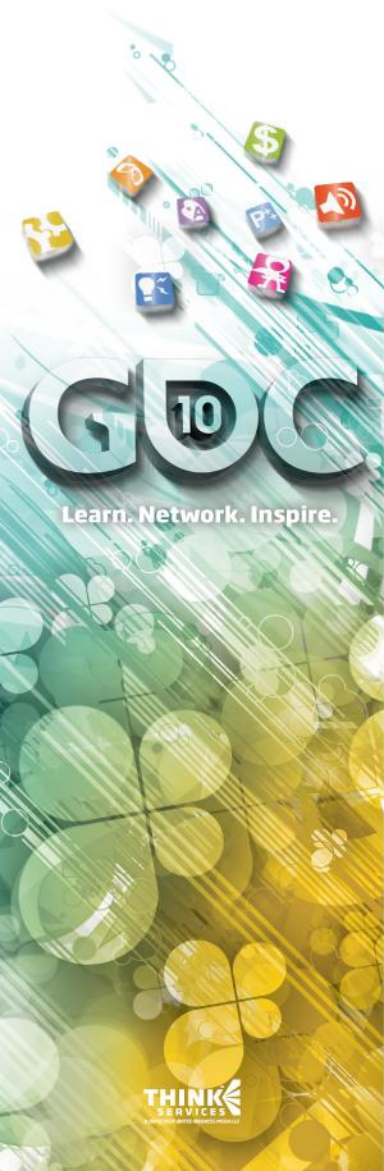
- ⌚ Non-Diegetic Audio
 - ⌚ Unnerving and Off-putting
 - ⌚ Adds Depth
 - ⌚ Builds Intensity



Sound Design

Creative Approach

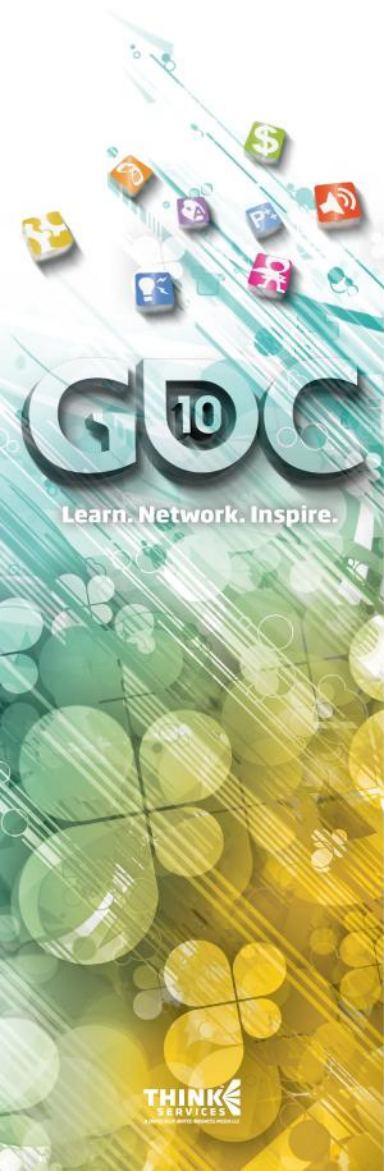
- ④ Cinematic Stylings
 - ④ If you think it's too big,
it's not big enough
 - ④ Hyper-realism
 - ④ Peaks and Valleys



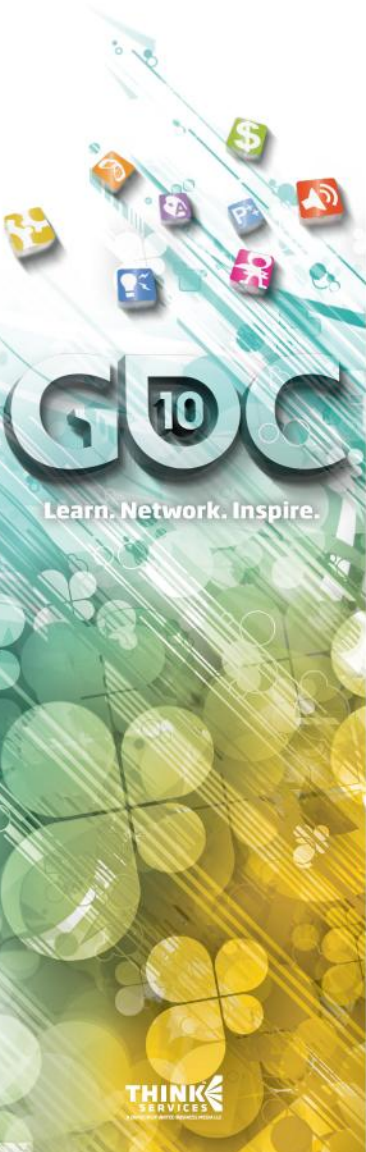
Sound Design

Mix States

- ⌚ Dynamic Mixing
 - ⌚ In-Engine Cut Scenes
 - Cinematic Moments
 - ⌚ Radio Ducking
- ⌚ Manipulation of FMOD's Plug-Ins

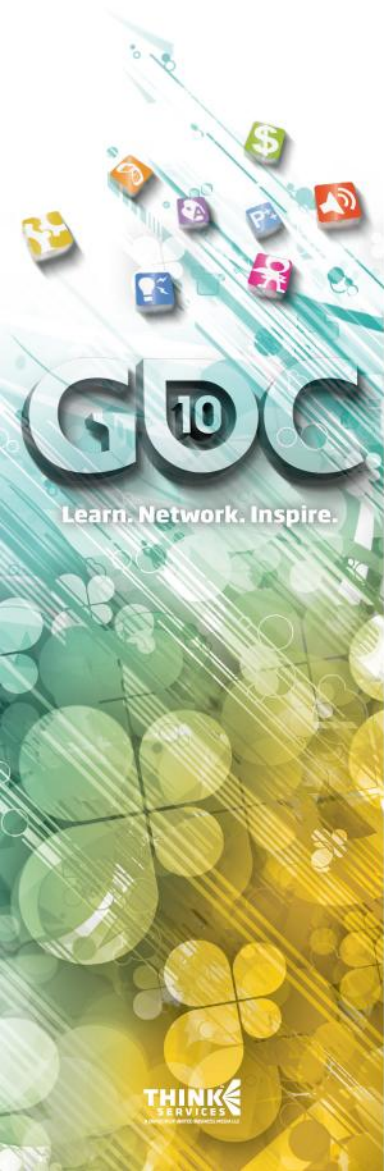


"All Things Are Possible"

- 
- ③ FMOD provides large chunks of functionality out of the box
 - ③ Advanced event editor
 - ③ Event Categories (aka Buses)
 - ③ "Et Cetera"
 - ③ Engine integration
 - ③ UnrealEd integration was already done
 - ③ ("But don't break it!")
 - ③ Lots more to do:
 - Reverb
 - Memory Management
 - Localization
 - Advanced Features
 - Debugging Tools
 - "Et Cetera"

Advanced Audio Features

- ⌘ Debugging
- ⌘ ~~Multi-Reverb~~
- ⌘ ~~Occlusion~~
- ⌘ ~~Virtualization~~
- ⌘ Background Sound Engine
- ⌘ Real-time Mixing Controls (Mix States)



Debugging



⌚ Goals:

- ⌚ Get Information

- ⌚ Improve Iteration Time

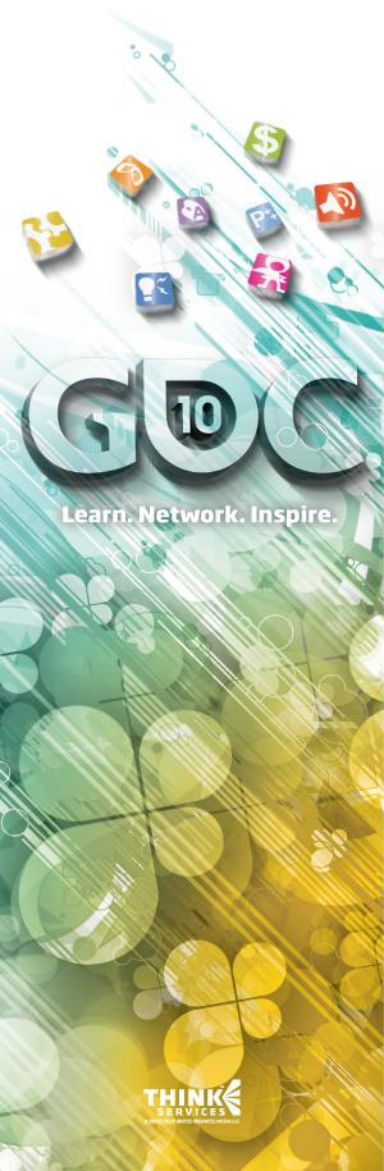
⌚ In-engine displays and logs

- ⌚ Mostly text-based

- ⌚ Meters, Spectrum, and source display

⌚ FMOD Designer Audition, ReloadAudioData, and FroAM

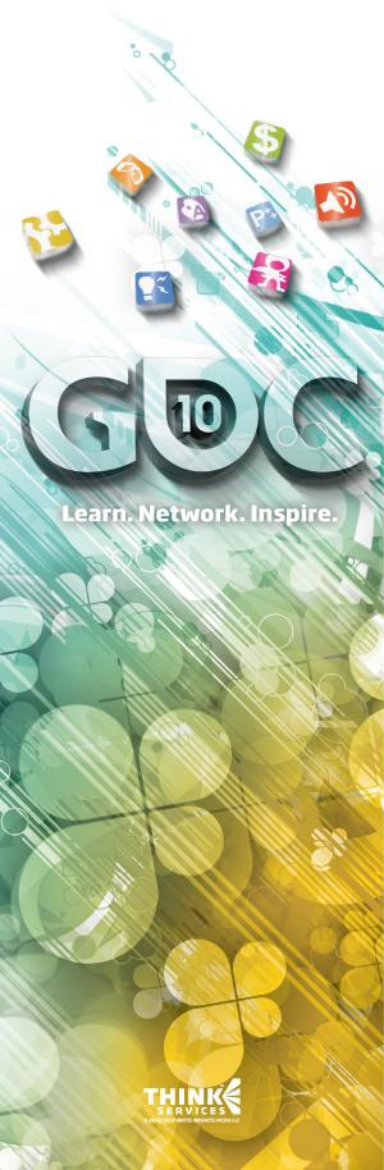
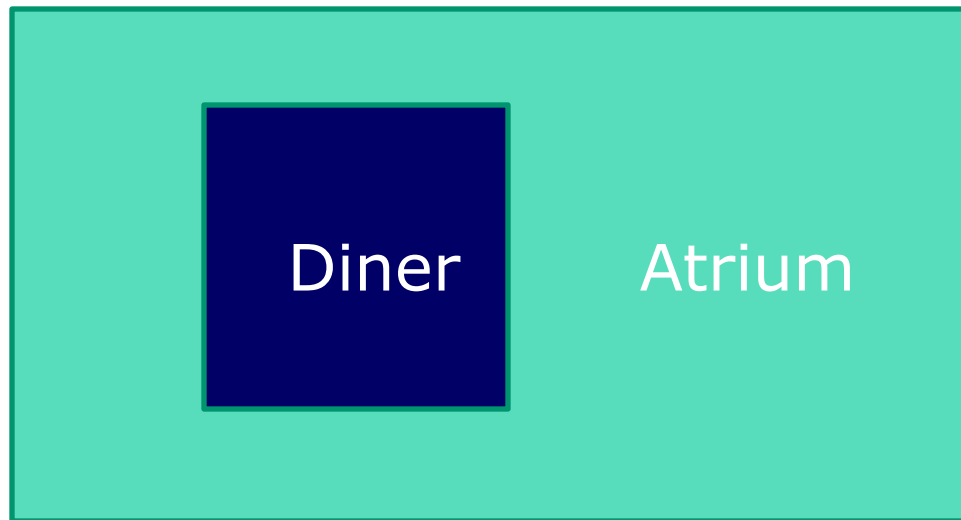
Debug Commands



⌘ BGSoundDebug	⌘ SoundDebugVirtual	⌘ StopSound
⌘ BGSoundToggle	⌘ SoundDebugVirtualizing	⌘ ToggleFMODDebug
⌘ BGSoundTogglePrimary	⌘ SoundGeometryToggle	
⌘ BGSoundToggleSecondary	⌘ SoundLoad	
⌘ BusDebug	⌘ SoundLoadPrefix	
⌘ ClearAllMixStates	⌘ SoundLogEventMemoryUsage	
⌘ ClearMixState	⌘ SoundLogLoadedMemoryUsage	
⌘ DumpTrackedMemoryInfo	⌘ SoundLogLoadedSounds	
⌘ FroAMTest	⌘ SoundLogLoadedSoundsAll	
⌘ KillLogs	⌘ SoundLogLoadedWaveBanks	
⌘ MixState	⌘ SoundMemoryDebug	
⌘ MixStateChaos	⌘ SoundMeters	
⌘ MixStateDebug	⌘ SoundMetersResetPeaks	
⌘ MixStateDebugAll	⌘ SoundMetersSpectrum	
⌘ OverridePrimaryReverb	⌘ SoundParamDebug	
⌘ OverridePrimaryReverbRoomEffectLevel	⌘ SoundPlayFilter	
⌘ OverrideSecondaryReverb	⌘ SoundSpamAllocations	
⌘ OverrideSecondaryReverbRoomEffectLevel	⌘ SoundStreamBankDebug	
⌘ PlayBGSoundPrimary	⌘ SoundStreamDebug	
⌘ PlayBGSoundSecondary	⌘ SoundToggleMemoryTracking	
⌘ PlaySound	⌘ SoundToggleOcclusionRender	
⌘ ReloadAudioData	⌘ SoundToggleReverbOcclusionRender	
⌘ ResetStreamStats	⌘ SoundToggleSourceDisplay	
⌘ ReverbDebug	⌘ SoundToggleSourceRadiusDisplay	
⌘ ReverbDebugSecondary	⌘ SoundUnload	
⌘ SoundChaos	⌘ SoundVirtualizeFilter	
⌘ SoundDebug	⌘ StopAllSounds	
⌘ SoundDebugAll	⌘ StopAllSoundsForce	
⌘ SoundDebugStopped	⌘ StopBGSoundAll	
⌘ SoundDebugToggle	⌘ StopBGSoundPrimary	
⌘ SoundDebugToPlay	⌘ StopBGSoundSecondary	

Multi-Reverb

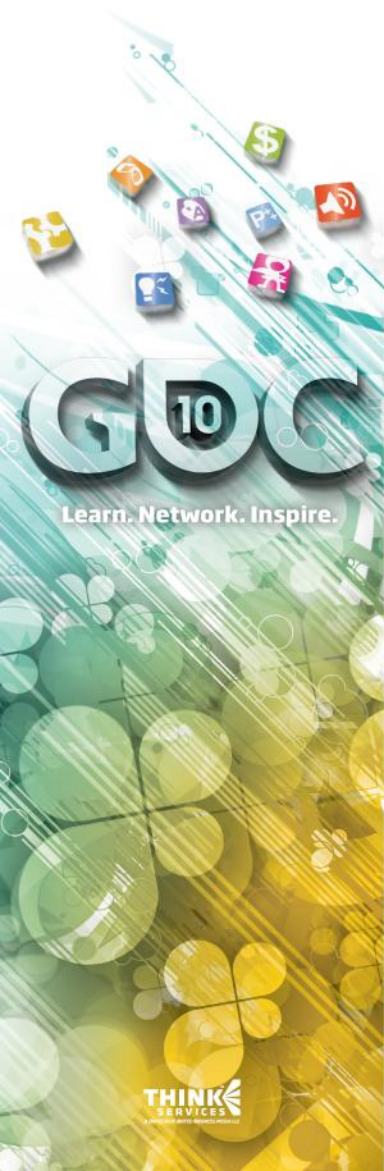
- ⌚ Reverb is reverb
 - ⌚ The middleware takes care of it
- ⌚ Problem:
 - ⌚ Pauper's Drop Diner



Multi-Reverb

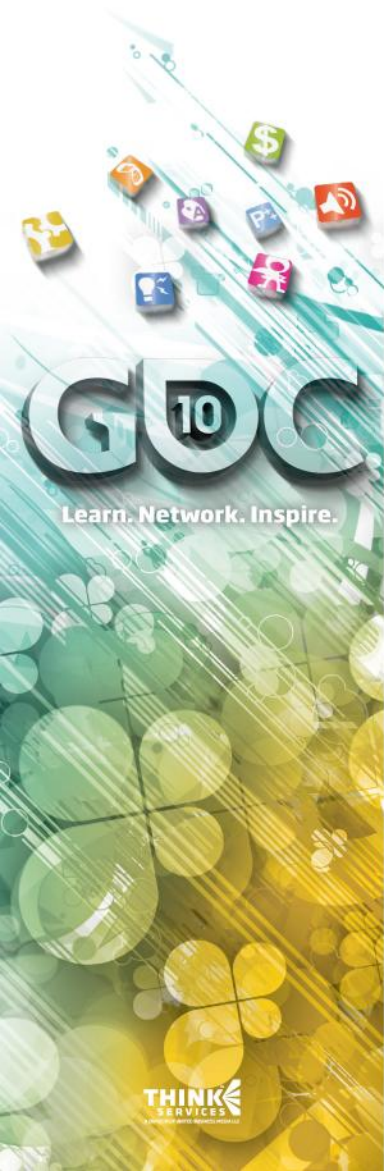
⦿ Solution:

- ⦿ Run two reverbs simultaneously
 - ⦿ (except on PS3 ☹)
- ⦿ Primary reverb is set by the zone that you are in
- ⦿ Secondary reverb is set by the nearest zone that has a different reverb
- ⦿ Sounds play in whichever reverb is most appropriate:
 - ⦿ Secondary if sound plays in a zone that has the secondary reverb
 - ⦿ Otherwise, Primary
- ⦿ Adjust room effect level on Secondary Reverb



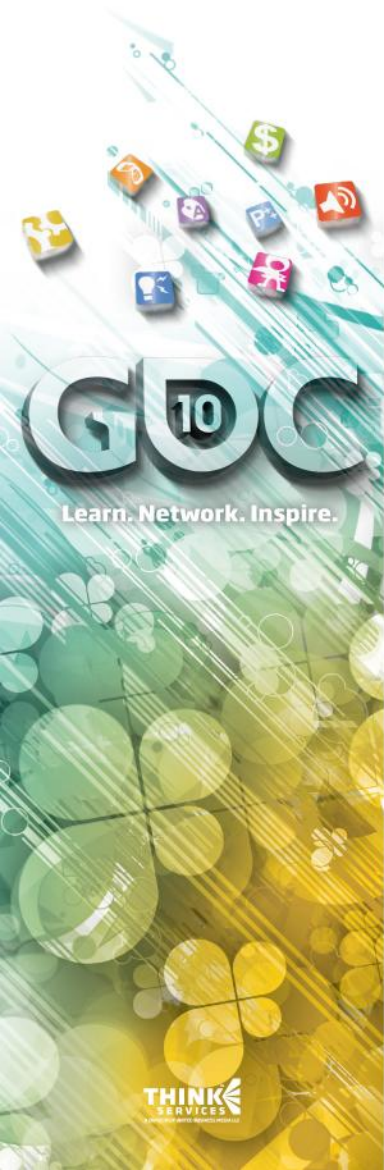
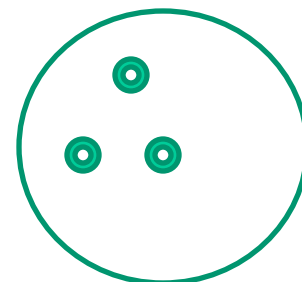
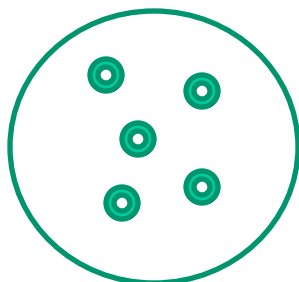
Occlusion

- ⊗ Old way in Bioshock 1: "Propagation"
 - ⊗ But Propagation != Occlusion
- ⊗ Mandate:
 - ⊗ Low CPU usage
 - ⊗ Ray casts are forbidden!
 - ⊗ WTF?
- ⊗ Solution:
 - ⊗ Cheat!
 - ⊗ We used FMOD's geometry engine
 - ⊗ Hand-edited occlusion values on collision mesh
 - ⊗ Lots of work, but it sounds great



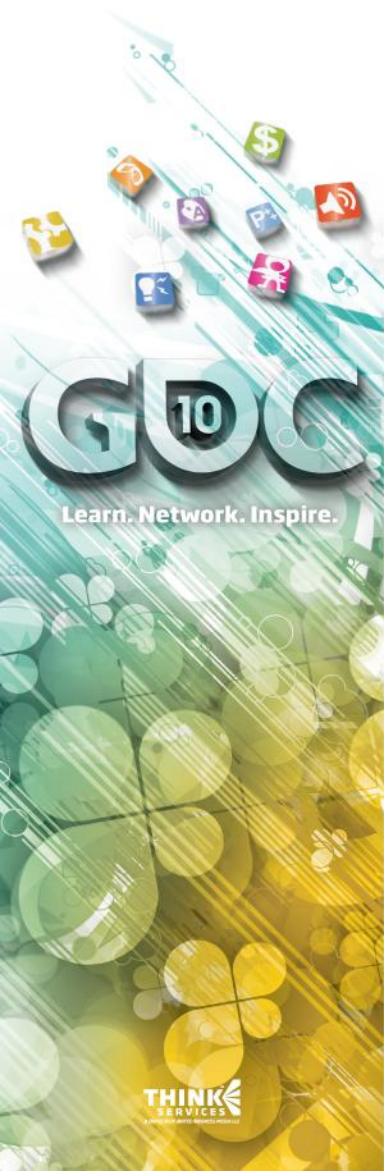
Virtualization

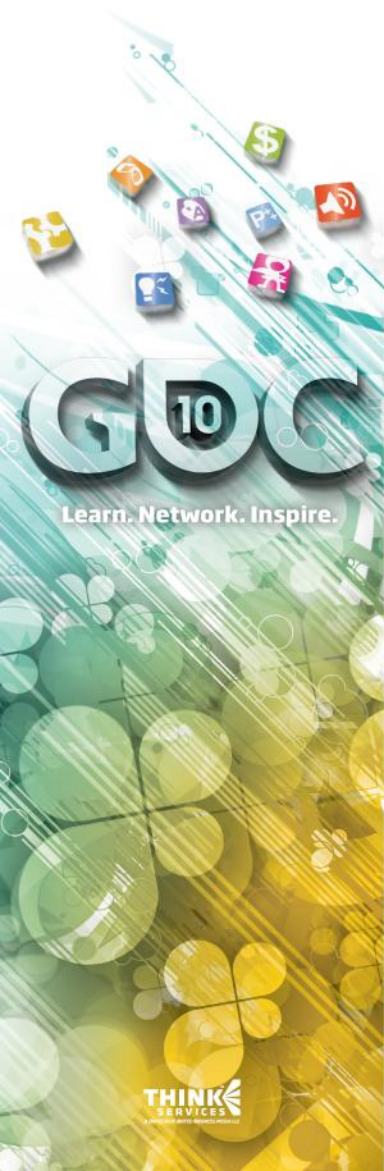
- ⌚ FMOD does virtualization for free
- ⌚ But we implemented our own
 - ⌚ More flexibility
 - ⌚ Cleaner implementation with respect to Event System
 - ⌚ Max Within Radius:



Background Sound Engine

- ③ My favorite!
- ③ A Background Sound Engine:
 - ③ 2D Sounds
 - ③ Loops that fade in and out, up and down over a random time
 - ③ 3D Sounds
 - ③ Sets of one-shots placed in a circle around the listener, move with the player
 - ③ Distance is faked with volume
 - ③ Limited by quadrant
- ③ We ran two engines (Even on PS3 😊)
 - ③ Primary by zone
 - ③ Closest dissimilar zone
 - ③ -6dB





Background Sound Engine Demo

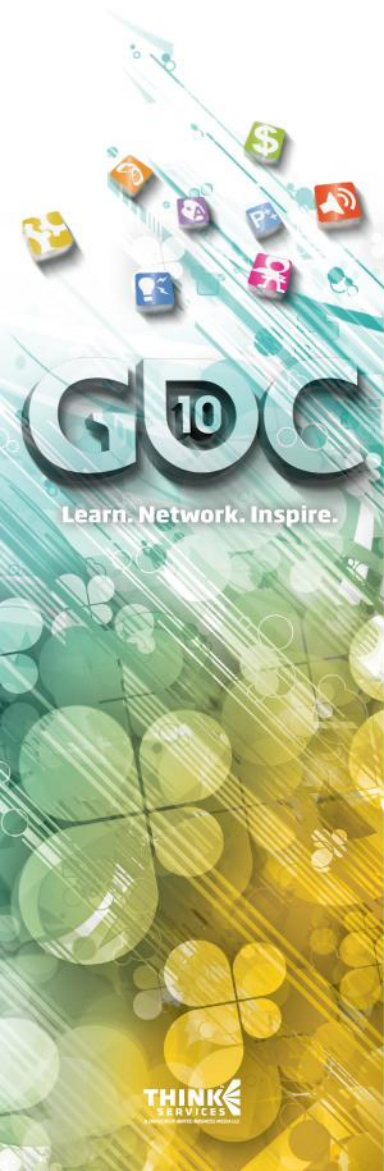
Mix States

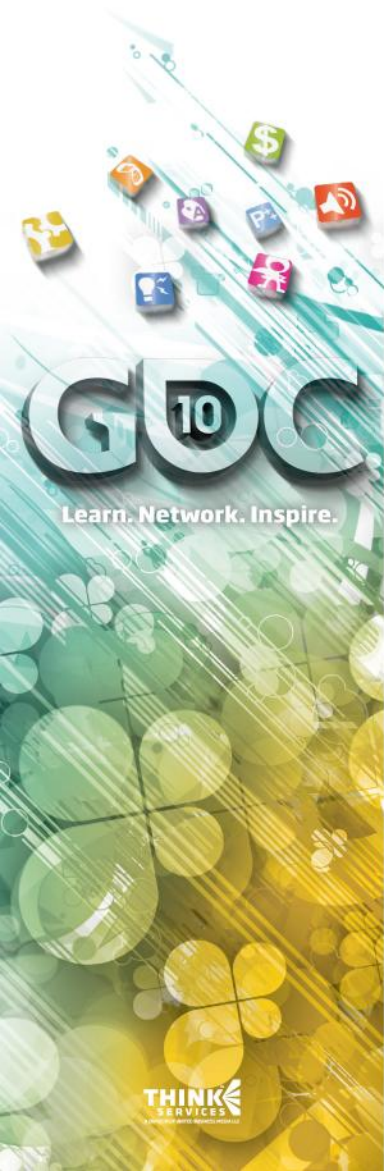
⌘ Triggers:

- ⌘ Sounds, Zones, Spheres, Scripts...

⌘ Controls:

- ⌘ Volume of buses with mix fades
 - ⌘ Increase or decrease volume, but no gain
- ⌘ Attach effect chains to buses
- ⌘ Override system reverb



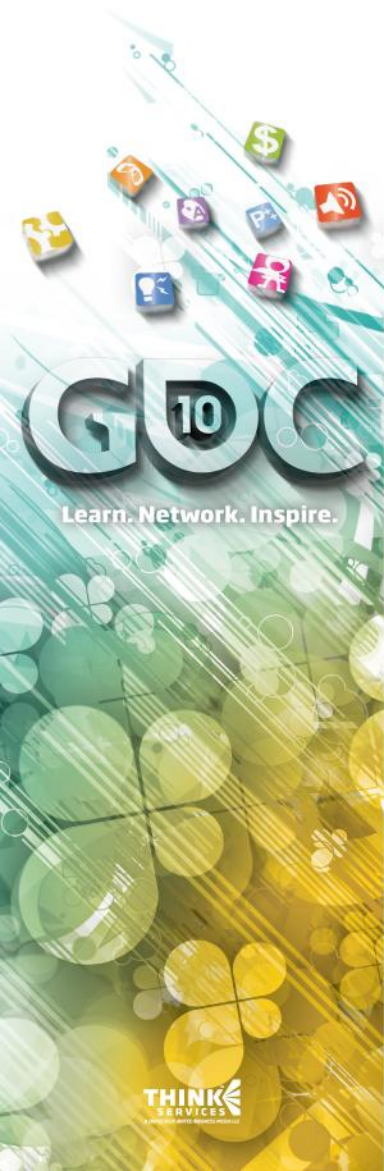


Mix State Editor Demo

Music

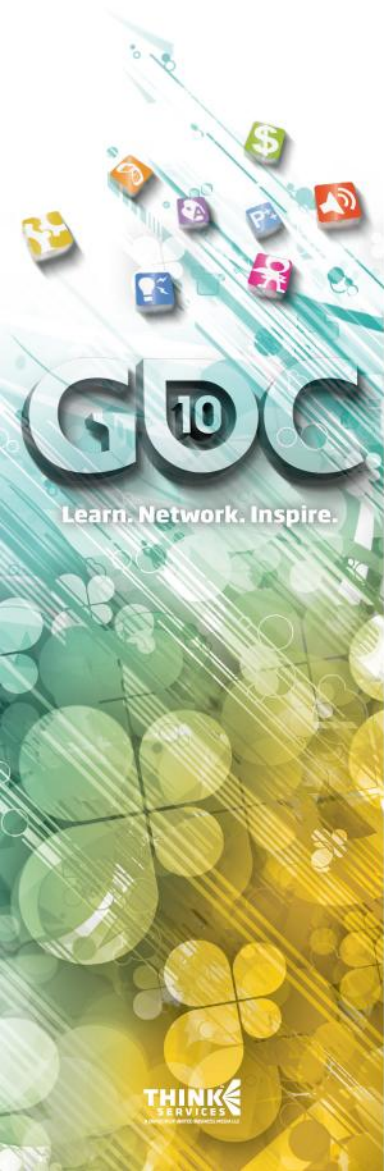
Music and Mix States

- ⌚ Combat Music Ducking
 - ⌚ Both Score and Licensed Tracks
- ⌚ Fading into Backgrounds
 - ⌚ Music is part of the ambience
- ⌚ Licensed Track Mixing
 - ⌚ Living believably in that space



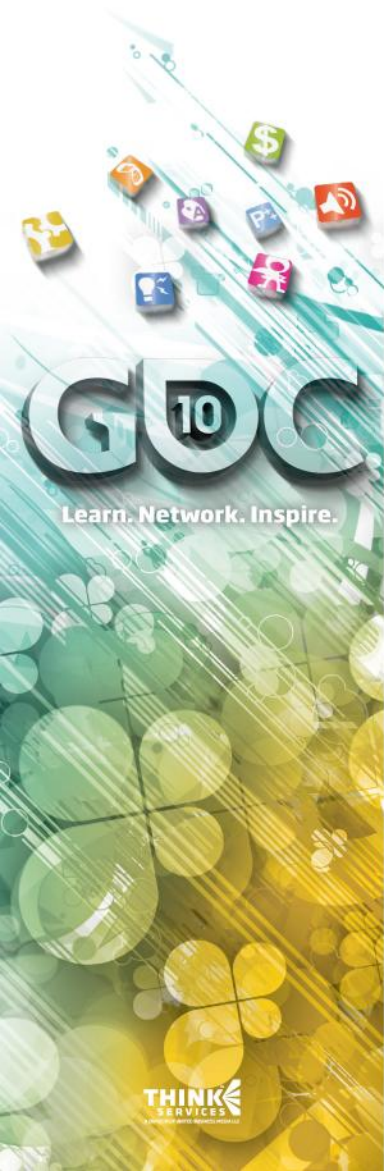
Music

- ④ Working with Garry Schyman
- ④ Creative Use of Source
 - ④ Using different tracks and mixes when thematically appropriate
- ④ Vary the Experience
 - ④ Each level has it's own musical signature
- ④ Basic Implementation
 - ④ No dynamic music system



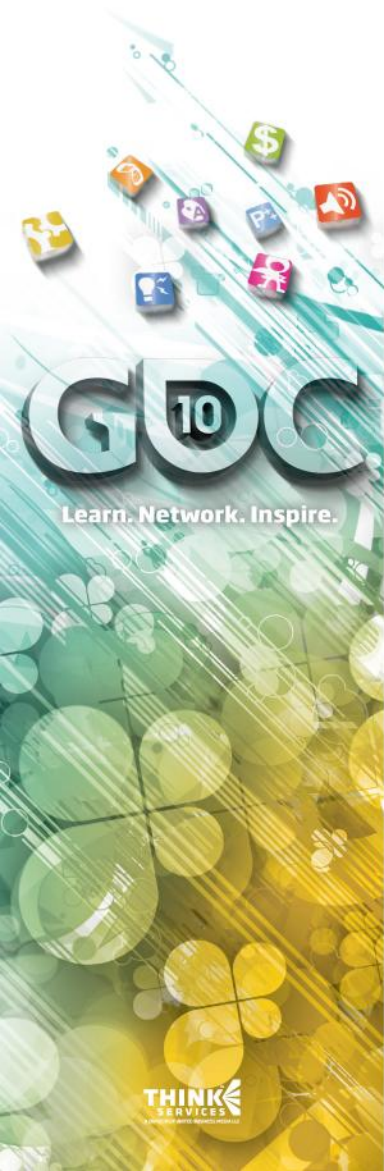
Dialogue Implementation

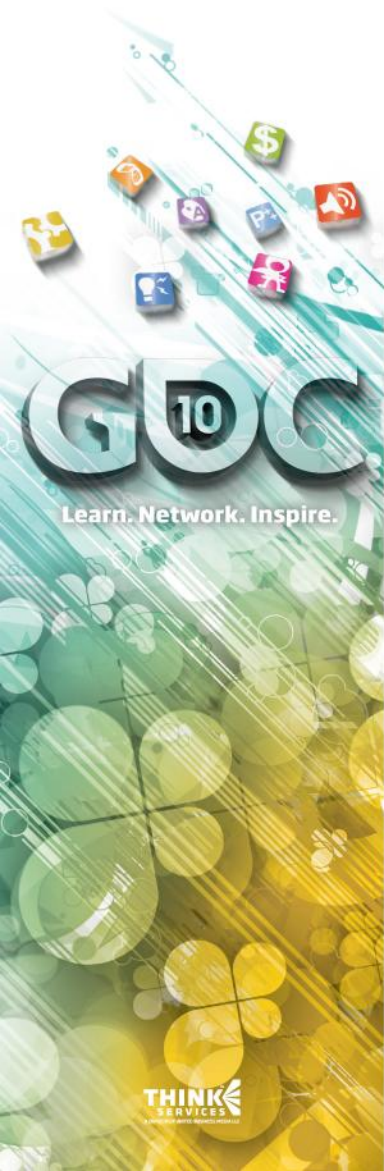
- ④ Sound designers built FSBs
- ④ Filename prefixes used to pick subsounds
 - ④ This is not best practice!
- ④ Problem:
 - ④ Too much content
 - ④ FSBankEx can be scripted, but still needs babysitting



Dialogue Implementation

- ③ Solution: FSB Builder
 - ③ C# app linked with C++ DLLs
 - ③ DLLs are thin wrappers for fsbanklibex
 - ③ **WARNING: fsbanklibex is not thread-safe!**
 - ③ FSB Builder uses three copies of the same DLL





FSB Builder Demo

Dialogue

④ Tools

④ covered

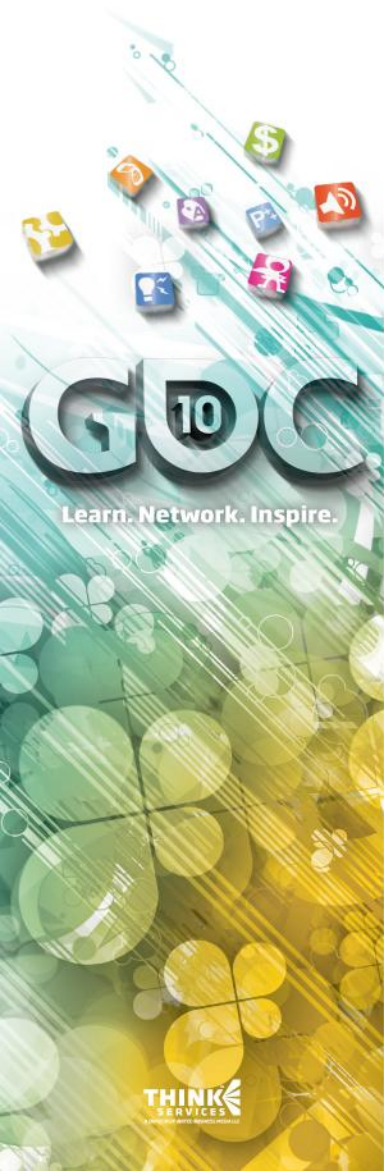
④ VO Process

④ PrePro

④ Production

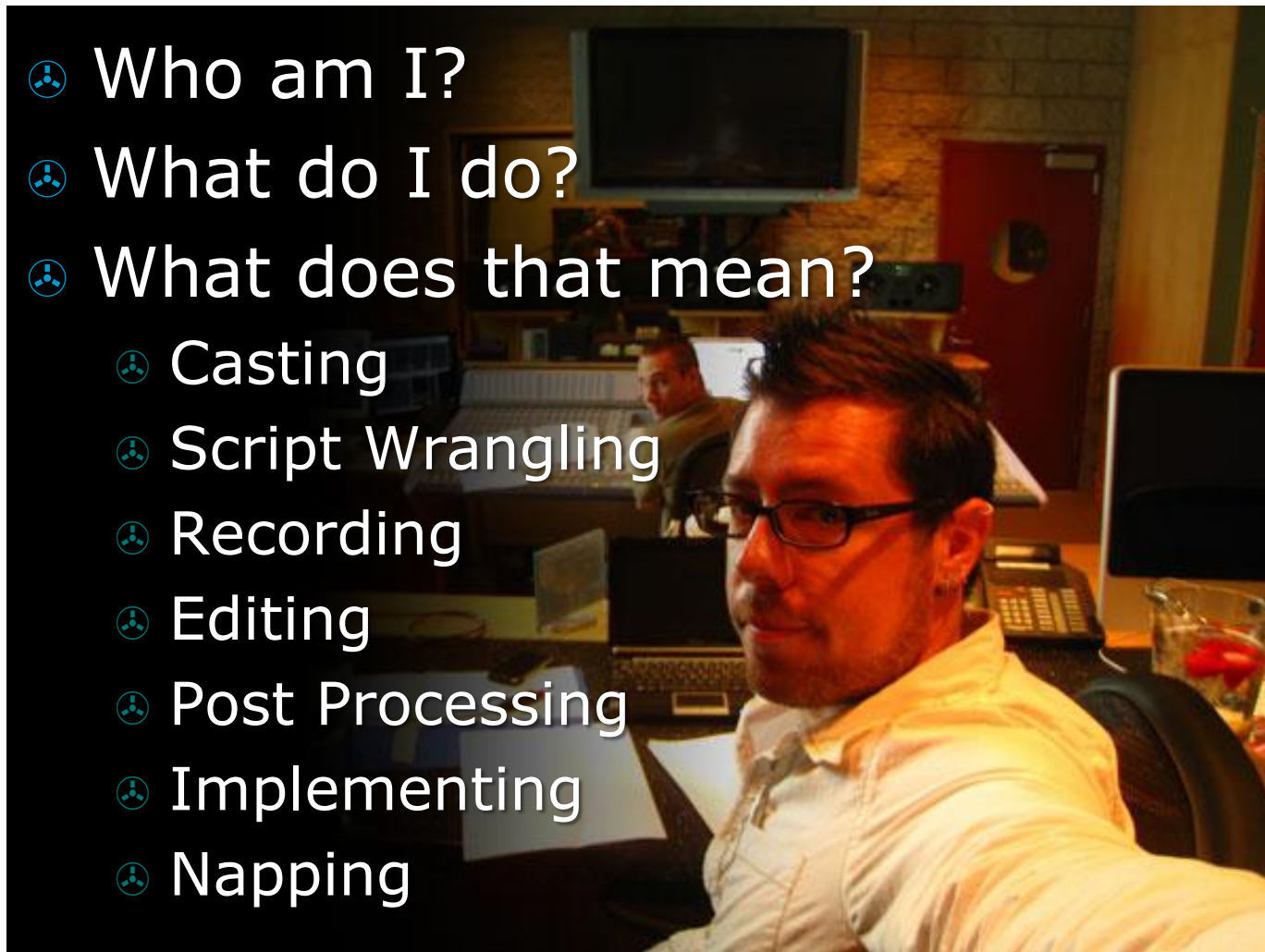
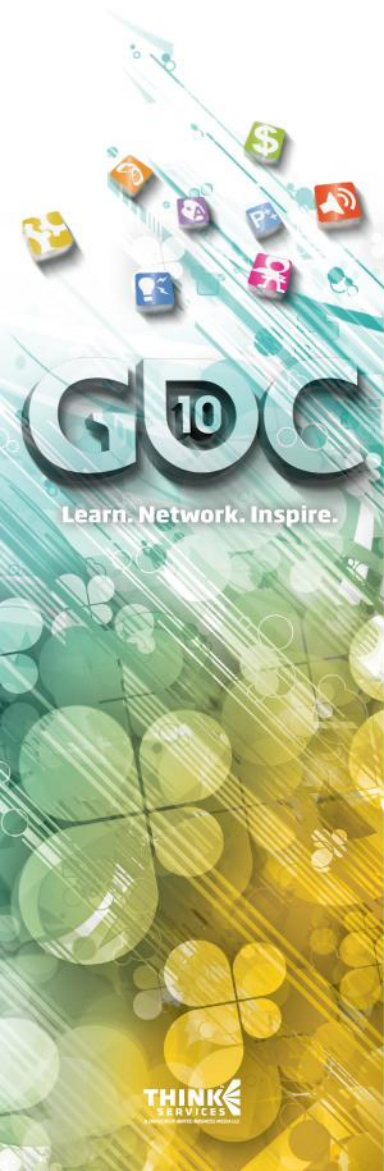
④ Post

④ Not so good -> Good



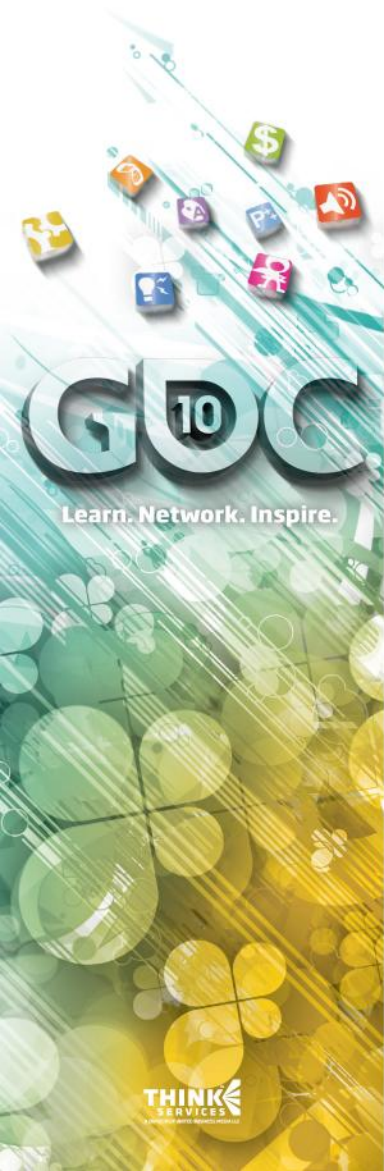
Dialogue

- ③ Who am I?
- ③ What do I do?
- ③ What does that mean?
 - ③ Casting
 - ③ Script Wrangling
 - ③ Recording
 - ③ Editing
 - ③ Post Processing
 - ③ Implementing
 - ③ Napping



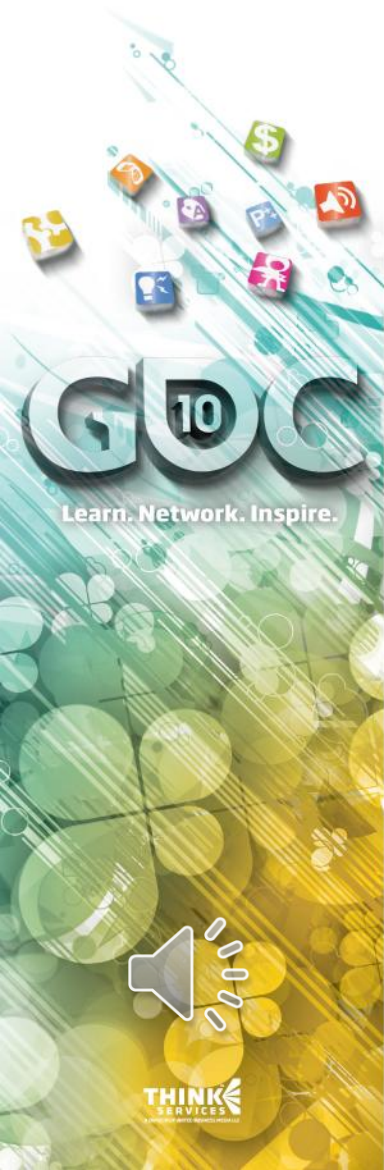
Pre Production

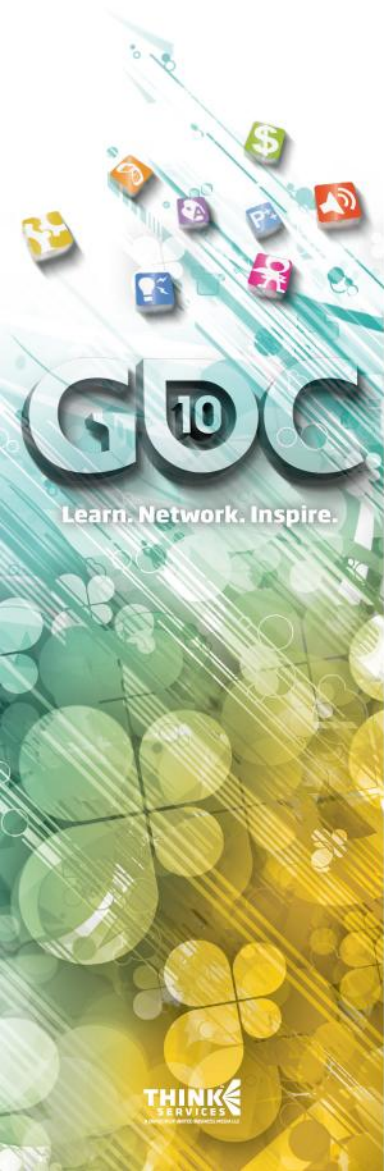
- ⌚ Didn't really have any
 - ⌚ No clear schedule
 - ⌚ No script review
 - ⌚ No tools assessment/development
- ⌚ Casting sides not prioritized
- ⌚ Professional crit-path scratch
 - ⌚ Real actors
 - ⌚ Better feel for the story
 - ⌚ More accurate focus testing
 - ⌚ It's how we found Sinclair
- ⌚ Precision Casting
 - ⌚ Unbelievable attention to detail
 - ⌚ Committed to accuracy



Production

- ③ Moving Milestones/Soft Deadlines
- ③ Incomplete Scripts
- ③ 2 Continents, 4 Countries, 6 Cities, 10 Studios
- ③ 30+ Crew, 70+ Cast (and their agents)
- ③ 30+ Crew, 70+ Cast (and their agents)
- ③ Large buckets of money
- ③ Hilarious Anecdotes





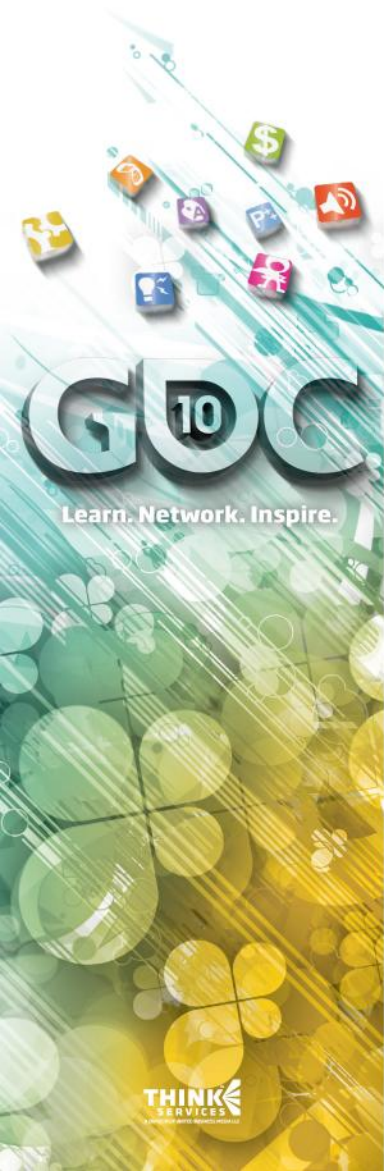
Sheryl Lee



Sheryl Lee Ralph

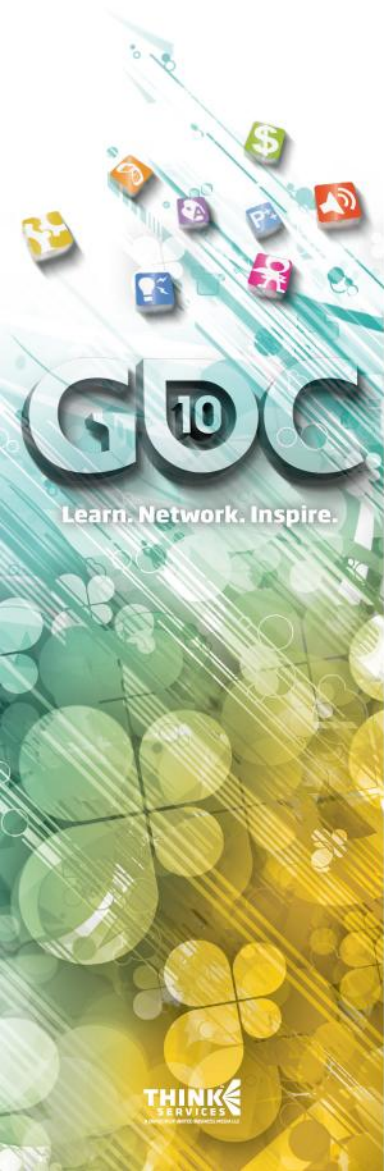
Post Production

- ⌚ Not enough time
- ⌚ No Dynamic Processing
- ⌚ Localization
 - ⌚ Big fat frowny face ☹
 - ⌚ Will forever now be mapped out diligently during preproduction
- ⌚ Editing
 - ⌚ Outsourced to Jason Kanter at Mako Audio
- ⌚ Implimentation
 - ⌚ Easy with the aid of Guy's tools
 - ⌚ I highly recommend getting a Guy



VO Summary

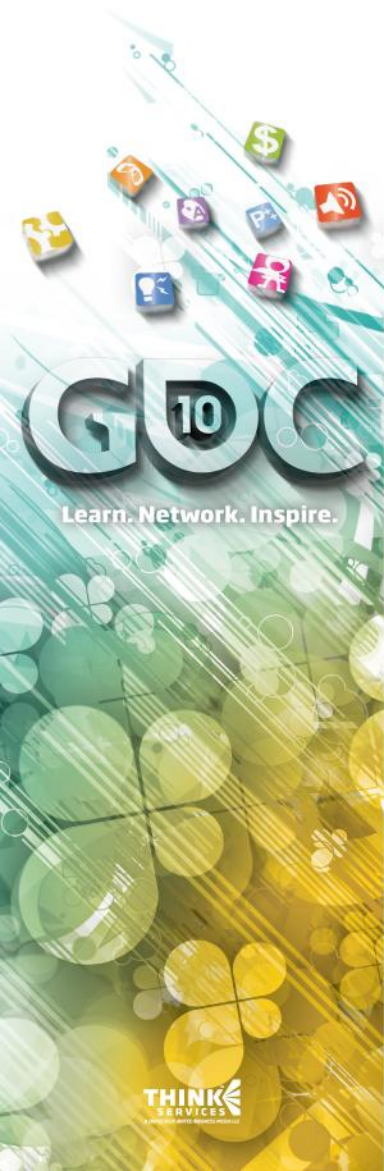
- ③ VO is a p1 dev pre-pro task
- ③ Pre-Pro is very, very important
- ③ Schedule backwards
 - ③ Start and localization and go to casting
- ③ Deadlines are deadlines
- ③ When booking talent, always write out the full name in all caps
- ③ Have fun with it



Wrap Up

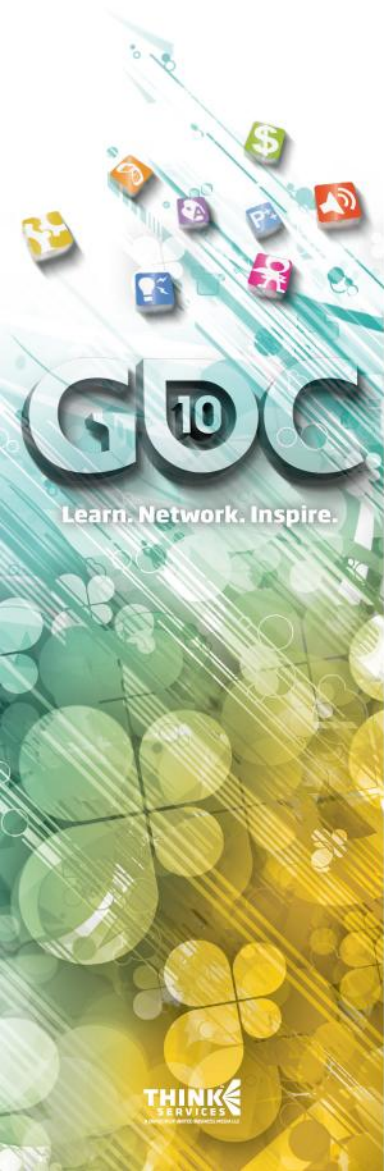
Lessons Learned

- ⌚ Pick the Right People
- ⌚ Find Your Tech Weak Spots Early
- ⌚ Have a Solid Vision
- ⌚ But Be Willing To Adapt
- ⌚ Always Push for What You Need

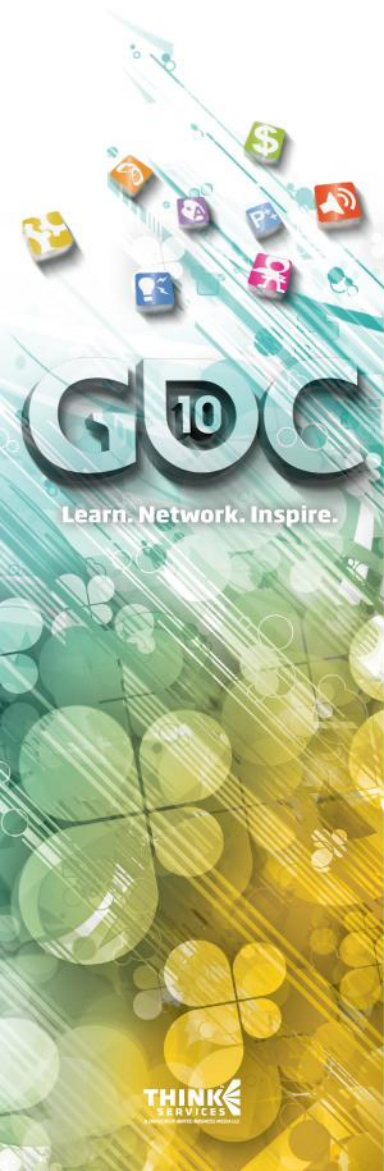


Four Classes of Audio for Media

- ③ Terrible Audio
 - ③ Sticks out, ruins experience
- ③ Bad Audio
 - ③ Detracts from experience
- ③ Good Audio
 - ③ Enhances experience
- ③ Great Audio
 - ③ Sucks player into experience



Any Questions?



Thanks For Attending!