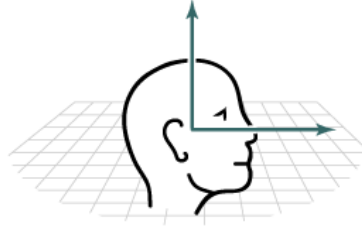# CS4455 – Audio for Games

Jeff Wilson

3D Audio (DS3D, OpenAL)

- Requirements:
  - Listener
    - Position
    - Orientation
  - Sound Source (e.g. point)
    - Position
    - Orientation (only if sound cone)

Interaural Timing/Intensity Difference (ITD/IID)

- With software, timing delays can be introduced to left/right channels
- Attenuation (volume adjustment) can also be applied per channel
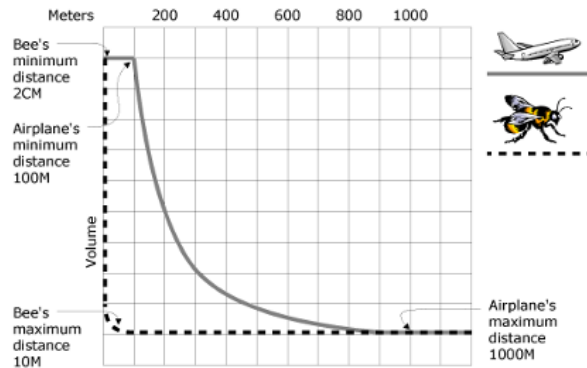- This helps create 3D effect

HRTF

- (Generalized) Head Related Transfer Function
- Effects of Pinna, head/skull, etc.
- Captured or generic
- Personalized from generic?
- Not usually a part of audio APIs

3D Audio - Source

- Min/max Dist
- Logarithmic Attenuation (Roll-off)
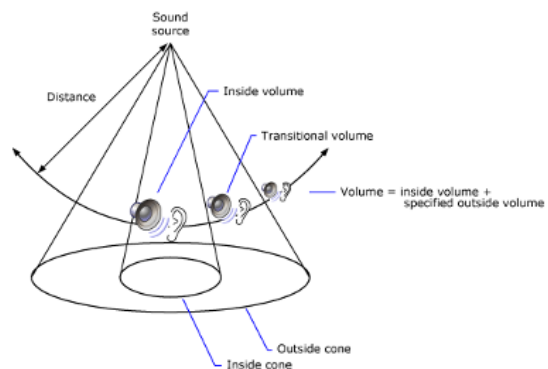- Custom attenuation models (e.g. linear, log with linear fade out)

3D Audio – Moving Sources

- Doppler
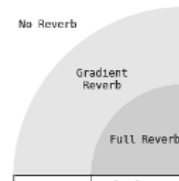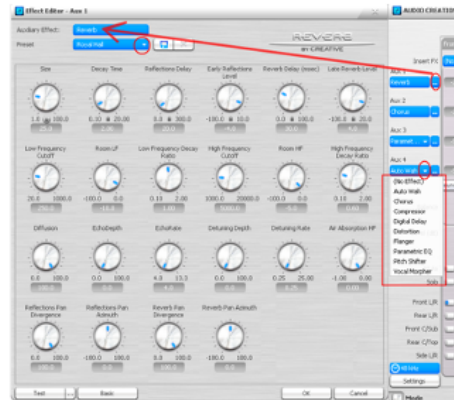- Can be exaggerated with scalar

3D Audio – Directional Source

- Sound Cone
- Orientation

3D Sound – Environment Effects

- Reverb
- Reflections
- Echo
- Band pass effects
- Transitions between areas
- Unity: Reverb Zone

Who Does Audio Design?

- Often musicians do both music and sound effects
- Also, audio engine programmers tend to create sound effects too
- Big budget projects more likely to have multiple audio specialists: musicians, audio engineers, foley artists, etc.
- Smaller budget projects will have overlap of job skills
- **If you want to specialize in audio in games, you need to either wear multiple hats or be the best in the business…**
- **…However, experience with audio programming / design is great for your resumé (may get you a job)**

Games Focusing on Music

- Music/Art Sandbox (Toshio Iwai)
  - Sim Tunes
  - Electroplankton
- Rhythm Games
  - PaRappa the Rapper
  - Amplitude / Frequency
  - Guitar Hero
  - Rock Band
- Singing
  - Karaoke Revolution (and other karaoke games)
  - Racing Pitch
- Music-Themed Interaction
  - Loom
  - Legend of Zelda: Ocarina of Time
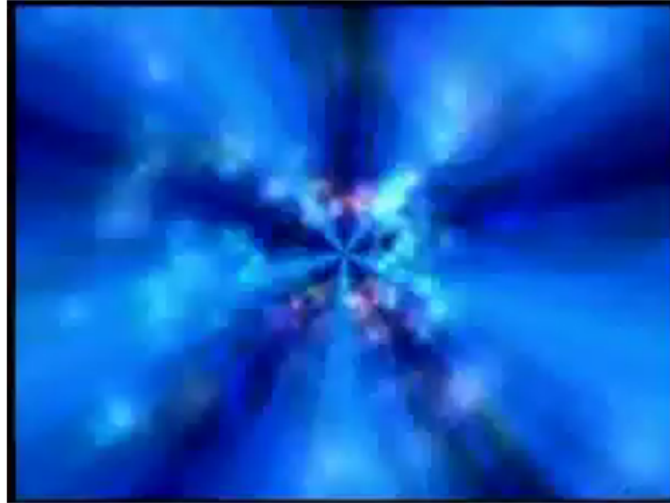
Sim Tunes

Toshio Iwai

Electroplanton
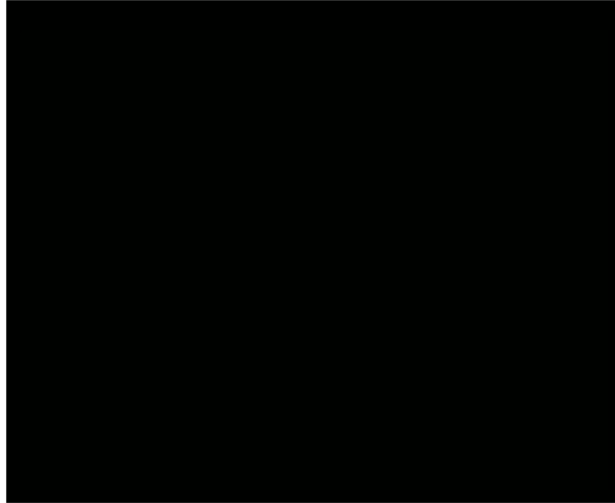
Toshio Iwai

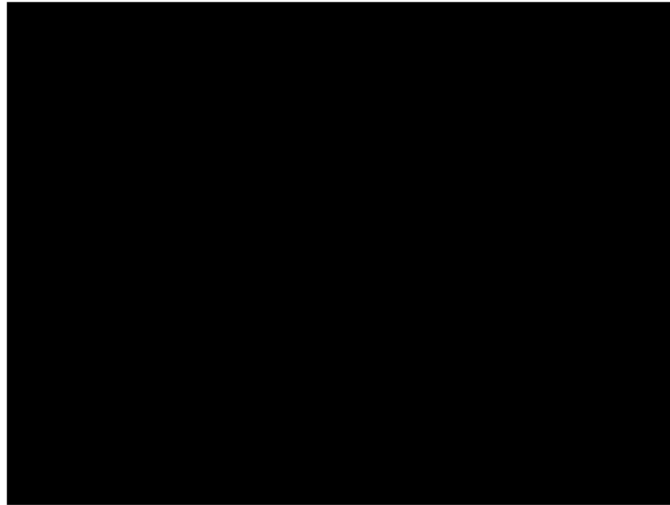# Parappa the Rapper

Amplitude

Racing Pitch

Loom

Games focusing on Audio

- Trespasser ("Foley Effects Engine")
- Thief
- You Don't Know Jack
- Papa Sangre (I/II)
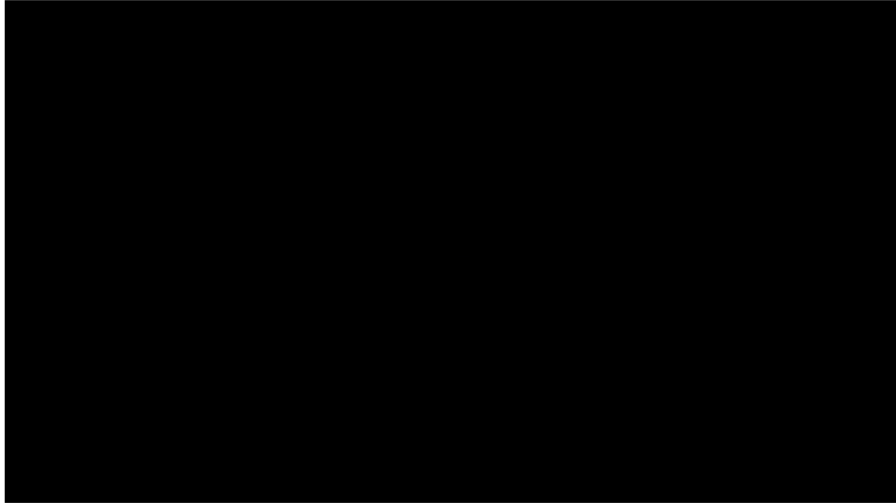- ???

Thief

You Don't Know Jack

Papa Sangre

Gambling Games / Classic Arcade Games

- Casino – pleasant sounds that all blend well together
- Arcade – synthesized sounds

Music Theory

- Based on string vibrations
- Harmonics – Dissonance – Consonance

Harmonics

- multiples of fundamental/root frequency (works for string instruments, resonating tubes, but not bells or gongs)
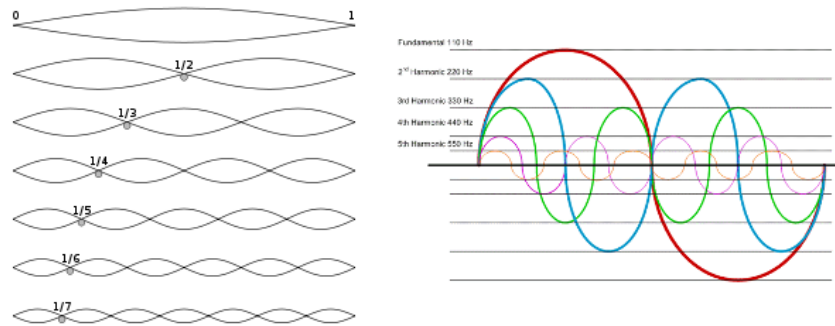
Dissonance

- unpleasant beating from overlapping frequencies

Wave 1 (red) and Wave 2 (blue)

Beat Pattern (in green)
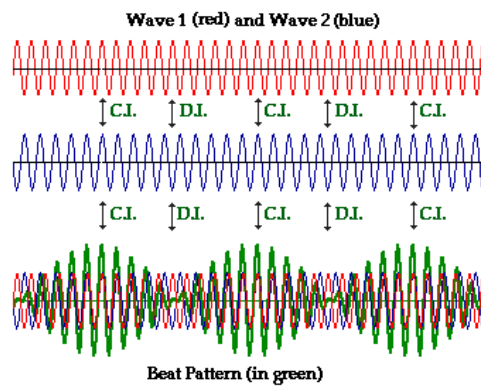
Consonance

- pleasant sounding mix of multiple frequencies (single string vibrations create a weak chord!)
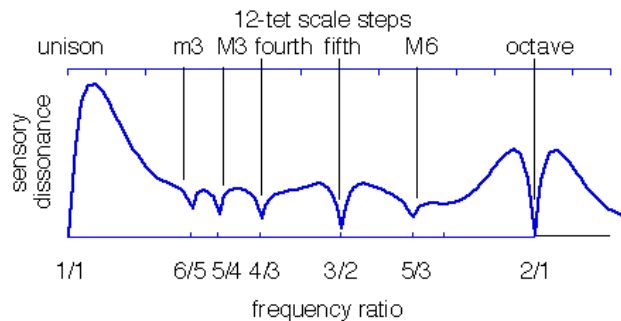
Assume C at far left and high C at almost right on "octave". G in the middle

Chords

- mix of generally pleasant sounding notes. Usually same volume level, and within an octave
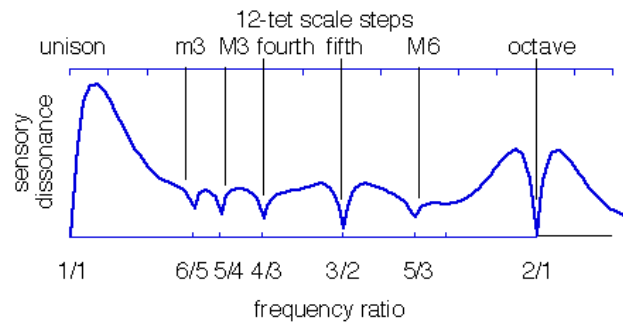
Look for dips on the "sensory dissonance" axis for candidate chords. C Major – C, E, G

Music Theory

- Music scale is a compromise
- Music theory from a mathematical/CS perspective:
  http://jjensen.org/musicTheory.html

- Synthesis can make it easier to create pleasant overlapping sounds (e.g. mute the harmonics that cause dissonance)

Audio Editing

- Editors
  - Basic cutting/pasting
  - Volume adjustment
  - Noise removal
  - DSP Filters/Effects
  - "Loop Tuner"
  - Synthesis
  - Multi-track editing (analogous to layers in Photoshop)

Popular Editors

- Basic
  - Audacity
  - GoldWave
- Pro
  - Sound Forge
  - Vegas (multi-track)
  - Acid
  - Pro Tools (emphasis on music)
- Synthesis
  - Max / MSP
  - Audio Mulch
  - ChucK – strongly timed, concurrent, and on-the-fly
  - Buzz (old school tracker / modular synth)
- MIDI
  - Cakewalk
  - CuBase

Audio Editing

- use high quality headphones or studio monitors in a quiet environment (otherwise you might miss a clip or pop)
- when recording live dialog/etc, also try to use headphones to monitor for clipping / distortion (don't find out after the fact that you've ruined sounds)
- Watch out for noise generators (fluorescent bulbs, computer fans, etc)

Audio Editing

- Log book / database
  - keep track of what you record, as well as location, equipment used, audio settings, voice talent, etc.
  - if you need to recreate similar sounds, it can be very useful to know what equipment you used before so you can recreate the same sound / background noise

Audio Editing

- Stitching sounds together
  - zero crossing
  - cross fade
  - allow discrete sounds to finish or it will stick out to user; easier to transition during continuous/droning sounds
  - example: shoot someone talking, interrupts them, then they scream

Audio Editing

- Loops
  - must link your loop end-to-end at same volume level to avoid pop (zero crossing generally easiest way to do this)
  - pay close attention to annoying repetitions. Even something like flowing water can be ruined if you choose your loop poorly
  - Some editors have "loop tuners"

Audio Editing

- Use of **Normalization**
  - allows rescaling of volume of audio signal
- **Audio level compression** / dynamic range compression (or DRC) / volume compression / limiting
  - useful when you want a roughly continuous volume level

Synchronization

- human perception less forgiving for audio anomalies than visual
- always synch video / 3d animation to audio and speed up/slow down/drop frames
- leave audio intact
- if you must adjust audio, use a real-time dynamic pitch correction (SOLA algorithm – synchronized overlap and add) because the human ear is very good at noticing changes in relative pitch

Audio Rendering

- Low level buffer management versus "play file"
  - low level complicated but allows you to stitch sounds/interrupt at known transition points
  - can be tricky to get low level code to work on all hardware (for PC)
  - "play file" simple/easier
  - Low level implementation detail buffer underrun (caused by not getting file read quickly enough):
    - loop on last chunk of buffer?
    - Insert silence?
    - Drag sound? (easiest to do this if you are decompressing a sound represented in frequency domain, then you can "hold" the frequencies – often used in video conference / streaming audio CODECs)

In Game Engines

- use audio file pre-loading if feature available
- otherwise, pre-play the sound for a fraction of a second at startup
- this avoids common problem of a delay on the first time the sound plays interactively