## Audio output

Audio events triggered by

- in-game events: a character animation -> footsteps; a weapon fires; an explosion
- sense of locale: waterfall, crickets chirping in park
- explicit scripting: a NPC speaks, a door slams shut

Mixing and rendering multiple audio sources has been done with hardware & APIs

- DirectSound -> Xaudio -> Xaudio2 (Microsoft platforms including Xbox, phone)
- OpenAL (OpenGL analogy)
- Platform-specific APIs have been part of SDK for eg PS3, Wii
- Commercial APIs (because easier to use? because easy to port?)

Increasingly, purely software mixing and rendering is used (Xbox360, Playstation3)



## Human hearing

Frequency range ~20Hz - ~20kHz;
every octave corresponds to doubling frequency
*Pitch* (perception of frequency) is not *frequency*:
a little perceptual squashing occurs at extremes

A pressure difference (amplitude) of 20 µpascals
is *defined* to be the quietest human-audible sound.
But frequency matters: max sensitivity 2kHz-4kHz
Amplitude is proportional to Power
Power is ~ proportional to loudness

Phons, Sones, Bels: *relative* measures
+ 10dB every tenfold increase of power
~ +3.01dB – ie $\log_{10}(2)$ – every doubling
A conversation making say "absolute" 60dB is
~ $2^{20}$ times louder than quietest audible sound

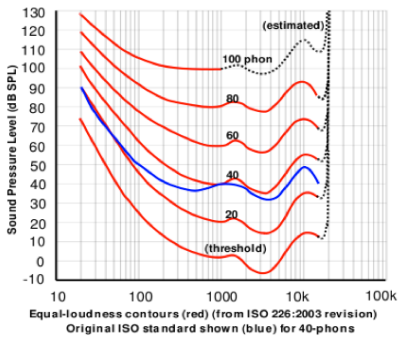
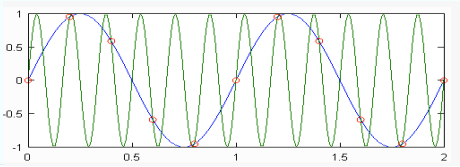



**Table 1: deciBel Ratings of Several Sounds**

| Sound Source | Intensity | deciBel |
|---|---|---|
| Weakest Sound Heard | $1 \times 10^{-12}$ W/m$^2$ | 0.0 |
| Rustling Leaves | $1 \times 10^{-11}$ W/m$^2$ | 10.0 |
| Quiet Library | $1 \times 10^{-9}$ W/m$^2$ | 30.0 |
| Average Home | $1 \times 10^{-7}$ W/m$^2$ | 50.0 |
| Normal Conversation | $1 \times 10^{-6}$ W/m$^2$ | 60.0 |
| Phone Dial Tone | $1 \times 10^{-4}$ W/m$^2$ | 80.0 |
| Truck Traffic | $1 \times 10^{-3}$ W/m$^2$ | 90.0 |
| Chainsaw, 1 m away | $1 \times 10^{-1}$ W/m$^2$ | 110.0 |

## Digital Sampling

- Sampling rate (CDs: 44100 samples/sec ), sample bit depth (CDs: 16 bits)
- There will be quantization error.
  - 16-bit samples offer 65536:1 SNR (Signal-to-Noise Ratio), ie ~96dB
  - Human hearing range is ~ 100dB
  - Improvements over CD quality almost undetectable
- *Nyquist* limit: A sampling rate can only represent frequencies up to half that rate
- The closer to theoretical maximum frequency, the worse the representation
  - That's behind CD choice of 44100: 44100 > 2 * 20k
- Beyond maximum:
  - like reversing cartwheels in movies






## Audio data pipeline

Permanent Storage
(Hard Drive/Optical Disk/ROM)

↓

(optional) Software Decoding
(MP3, Ogg Vorbis)

↓

Memory buffering

↓

Sound Channel Processing
(2D or 3D processing effects)

↓

Hardware mixing of all channels and
digital-to-analog signal conversion

from
*Introduction to Game Development*
(2nd ed), Steve Rabin

## 2D and 3D processing

- Pan operation:
  - with mono source: for 2D, adjust stereo by attenuating left or right
  - for 3D, also attenuate both left and right to simulate distance
  - *but:* audio environments differ: laptop speakers, earphones, surround sound
- Pitch control:
  - simple method: process more or fewer samples per second
  - this simple method has bad side effect of altering playback time of sample
- Volume control:
  - attenuate appropriately: remember -3dB means halve the power
  - interpolation easier with a linear scale (eg power) than a log scale (eg dB)



## Audio Compression

Bit-reduction schemes

- ADPCM – (Adaptive Delta(or Differential) Pulse Code Modulation)
- ADPCM and PCM are subclasses of Microsoft's WAV format
- ADPCM achieves fixed 4:1 compression, is simple to decompress
- Used on PSP, Wii, Nintendo DS (Dual Screen/ Developers' System)

Psycho-acoustic schemes

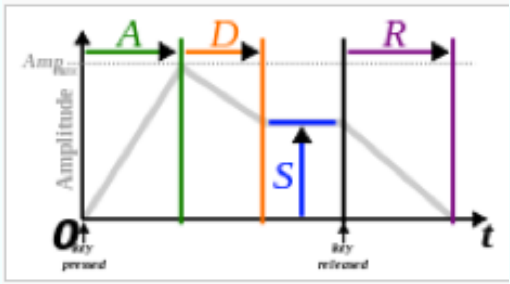
- MP3; also Ogg Vorbis, Microsoft's Windows Media Audio, Sony's ATRAC3
- Lossy schemes that throw away detail people could hardly hear anyway
- Can be parameterised, achieving 5:1 – 25:1 compression
- 10:1 compression readily achievable on CD-quality audio
- Saves disk space, memory: used in PS3, Xbox360, PC

MP3 is patented, licence fees payable; Ogg Vorbis is open-source and licence-free

## ADSR envelopes

Just 4 numbers required: Attack – Decay – Sustain – Release

- Originally developed for real-time environment – musical synthesizers
- Sustain period ends when key is released, so no need for Sustain duration
- If used for a note in a music score, you will also need Frequency & Sustain period




Schematic of ADSR



## 3D audio

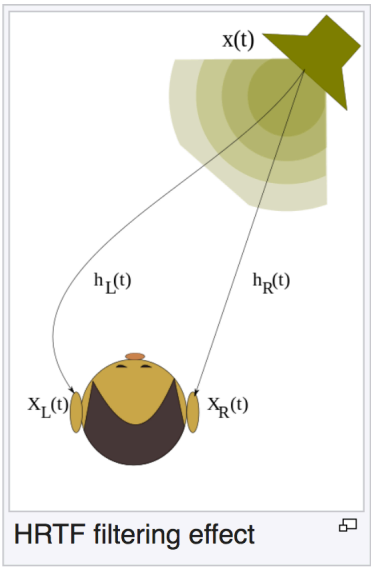
Use understanding of how brain uses aural cues to locate source of sound signal

- slight timing difference left ear/right ear
- ear shape causes attenuation of higher frequencies of sounds behind head

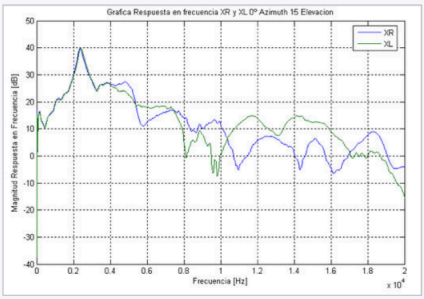
- with HRTF (Head Relative Transfer Function)
  - two speakers/earphones can mimic 3D sound origin quite successfully
    - especially with earphones
  - but it can be frustrated for several reasons
    - variety of consumers' speaker arrangements (eg surround sound)
    - room in which sound is played
    - individual's ears

Not really game developer's problem. Developer must provide data for sound mixing:

- Define sound source: position, velocity (for Doppler), cone angles
- Define listener: position, orientation. (*Listener space* is similar to *Camera space*)




HRTF filtering effect




A sample of frequency response of ears:

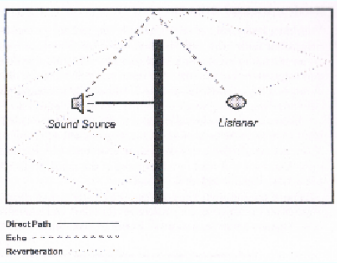
- *green curve*: left ear $X_L(f)$
- *blue curve*: right ear $X_R(f)$

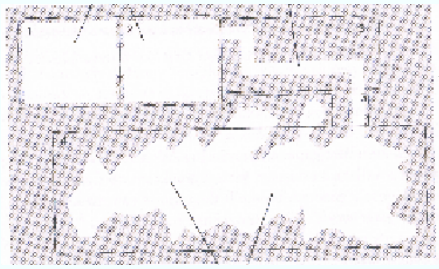
for a sound source from upward front.



## Effects of environment

Sound travels: directly; by early reflection (echo); by late reflection (reverberation)



both from *Introduction to Game Development* (2nd ed), Steve Rabin

- Room geometry may cause *obstruction* of direct transmission, and limit reflections
- Materials (fabric, stone) will affect reflections in different ways – *diffusion*
- Materials also may affect direct transmission – *occlusion*

Two standards:

- I3DL2 (Interactive 3D Audio Rendering Level 2) (interactive audio SIG)
- EAX (Environmental Audio Extensions) (Creative Labs)

## Sample-based audio player: MIDI and others

MIDI = Musical Instrument Digital Interface

- Encodes a passage of music using musical notation not as recorded sound
- Individual instruments have to be sampled; these samples are played back
  - repertoire of instruments may be limited; but control of playback can be gained
    - switch instruments; change key; adjust tempo; synchronise to a beat
    - low storage requirement (useful for GameBoy, Nintendo DS, downloads)

DLS (DownLoadable Sound) – format that packs instrument samples with MIDI

iXMF (Interactive Extensible Music Format) –

- provides further packing of waveforms & metadata
- expected to be useful for games with interactive music system



## Digital Audio Stream Player

Digital Audio Streams are merely recordings of music

- easy to create, easy to play
- storage hungry
  - can consume large fraction of game's memory
- hard to manipulate meaningfully, except for looping & sequencing

With either digital audio streams or MIDI-style players, short passages of music can be put in sequence, either with random branching or (if indexed) with thematic cohesion.

## Audio Scripting

- Eliminate programmer concerns with sound files: instead, trigger audio scripts.
- Audio scripting language should provide for common issues in game audio
  - Sound variation
    - eg footsteps may vary, randomly, or depending on ground type
  - Sound repetition
    - use sounds many times over, overlaid, but with limit (sword clanks)
  - Complex looping
    - start-loop-stop sequence eg for lifts, other machines
  - Background ambience
    - combine many elements, some looping & some not, into *soundscape*
    - random change of volume, pitch, timing of some elements –good for wind



## Voice and Language: Challenges

- lip-sync
  - opening/closing mouth as sounds are louder/quieter performs quite well
  - or, process transcript of performance to identify syllables or phonemes
  - or, analyse the sound itself (the advantage is language-neutrality)
- commentary
  - too many combinations of events eg in sports games
  - so stitch together small phrases while maintaining intelligibility & fluency
  - hard to put player-chosen names into commentary without jerkiness
- voice recognition
  - allowing player to speak from small repertoire of distinct commands or options