MPEG-4
Structured Audio

Eric D. Scheirer
MIT Media Laboratory
Editor of MPEG-4 Audio standard
eds@media.mit.edu
MPEG-4 Structured Audio

- General-purpose software synthesis
  - New algorithmic synthesis language: SAOL
  - Based on years of computer-music research
- Wavetable synthesis
  - Using DLS-2 format (“SASBF”)
- Streaming MIDI control
  - Also more flexible control with SASL
- TTS Interface
Music synthesis in MPEG?

• General-purpose synthesis not only for music
  – sound atmospheres, Foley effects, post-production, as well as music
  – very, very low bitrates

• Standardization encourages more/better implementations

• Drive forward the world of PC sound
Structured Audio model

Bitstream Header → SAOL Decoder → Reconfigurable Synthesis Engine

Wavetables

SASL/MIDI Decoder → Control

Multichannel high-quality audio
SAOL

- Structured Audio Orchestra Language (pronounced “sail”)
- New “Music-N” language
- Historically derived from Csound, but redesigned from ground up
- Formal language design
- Exact sound quality
- Reference implementation (slow) by MIT
Music-N model

- Invented M. Mathews early 1960’s
  *(Earliest synthesis technology)*
- *Orchestra* made up of *instruments*
- *Score* gives performance parameters
- No fixed instrument parametrization
- Instruments use built-in *opcodes*, plus arithmetic, to calculate DSP
Music-N Model

Execution

Fetch/Store

Wavetable 1
Wavetable 2
Wavetable 3
Wavetable 4

Instantiation

I1
I2
I3

SCORE

Performance
Simple SAOL instrument

instr beep(pitch, amp) {
    ksig vibfreq, env;   // control signals
    asig sound, f1, f2;  // audio signals
    table vibshape(harm, 128, 1);  // sine wave

    vibfreq = cpsmidi(pitch) * (1 + koscil(vibshape, 5)/40);
    sound = buzz(vibfreq, 0, 1, 0.9);  // noisy sound source
    f1 = bandpass(sound, 500, 100);   // formants
    f2 = bandpass(sound, 700 + vibfreq*2, 100) / 10;

    env = kline(0, 0.05, 1, dur, 0);
    output((sound + f1 + f2)*amp*env);
}

## Stuff built into SAOL

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math functions</td>
<td>Karplus-Strong synthesis</td>
</tr>
<tr>
<td>Granular synthesis</td>
<td>Fractional multi-tap delay lines</td>
</tr>
<tr>
<td>OLA FFT/IFFT</td>
<td>Arbitrary FIR, IIR filters</td>
</tr>
<tr>
<td>Reverb, chorus</td>
<td>Parametric compressor-limiter</td>
</tr>
<tr>
<td>Parametric filters</td>
<td>Several noise generators</td>
</tr>
<tr>
<td>“Bus routing” FX model</td>
<td>Pitch- and time-shifting for samples</td>
</tr>
<tr>
<td>Window functions</td>
<td>User-extensible language model</td>
</tr>
<tr>
<td>Graceful degradation</td>
<td>Dynamic note spawning</td>
</tr>
<tr>
<td>Deep connection to SASBF (DLS-2) synthesis</td>
<td></td>
</tr>
</tbody>
</table>

and more...
DLS-2 Wavetable Synthesis

- DLS-2 a harmonization of
  - SoundFont from E-Mu
  - Downloaded Sounds from MMA
  - One standard promotes product compatibility

- Entire DLS-2 synth embedded in SAOL
  - Write new filters/envelopes in SAOL
  - Filter MIDI input in SAOL
  - Generate new notes in SAOL
Text to Speech (TTS)

- *Interface* only
  - synthesis engine outside of standard

- Specifies
  - text
  - prosody and duration
  - facial animation cues
Example

• “Manipulator” (Michael Casey)
  – 84 seconds long (14.4 MB as WAVE file)
  – 16 KB in header
  – 11 KB in bitstream (= 0.13 kbps)
  – 100 KB in compressed samples

⇒ more than 100:1 compression, no loss of quality
Example

- “Xanadu” (Joseph Kung)
  - 60 seconds long (10.5 MB as WAVE)
  - 2.2 KB in header
  - 4.2 KB in bitstream (= 0.07 kbps)
  - no samples anywhere

⇒ more than 1200:1 compression, no loss of quality
Example

- *Le Sacre du Printemps* - Stravinsky
  (sound design by Andrew Horner)
  - 47 seconds (3 MB as WAVE)
  - 50 KB in header
  - 18 KB in bitstream ( = 0.5 kbps)
  - no samples, only synthetic sound

⇒ more than 40:1 compression,
no loss of quality
Public SA projects underway

- SAOL-to-C JIT translator
  - Soon SAOL-to-DSP, maybe SAOL-to-FPGA
- Real-time SAOL interpreter
- Web-embedded SA (like Beatnik)
- Better SAOL compiler tools
- Csound-SAOL-Csound translation
- “Visual orchestra” authoring tools
Summary

• MPEG-4
  high-quality, low bitrate, many tools

• Structured Audio
  streaming synthetic sound
  very flexible, MIDI-aware, high-functionality
  TTS interface

• International Standard
  will be supported by many companies
For more information

- MPEG-4 Structured Audio homepage
  http://sound.media.mit.edu/mpeg4/

- Several papers in technical literature
  *Multimedia Systems* 7:1 (Jan 1999)
  105th AES proceedings (Sept 1998, #4811)
  more forthcoming...