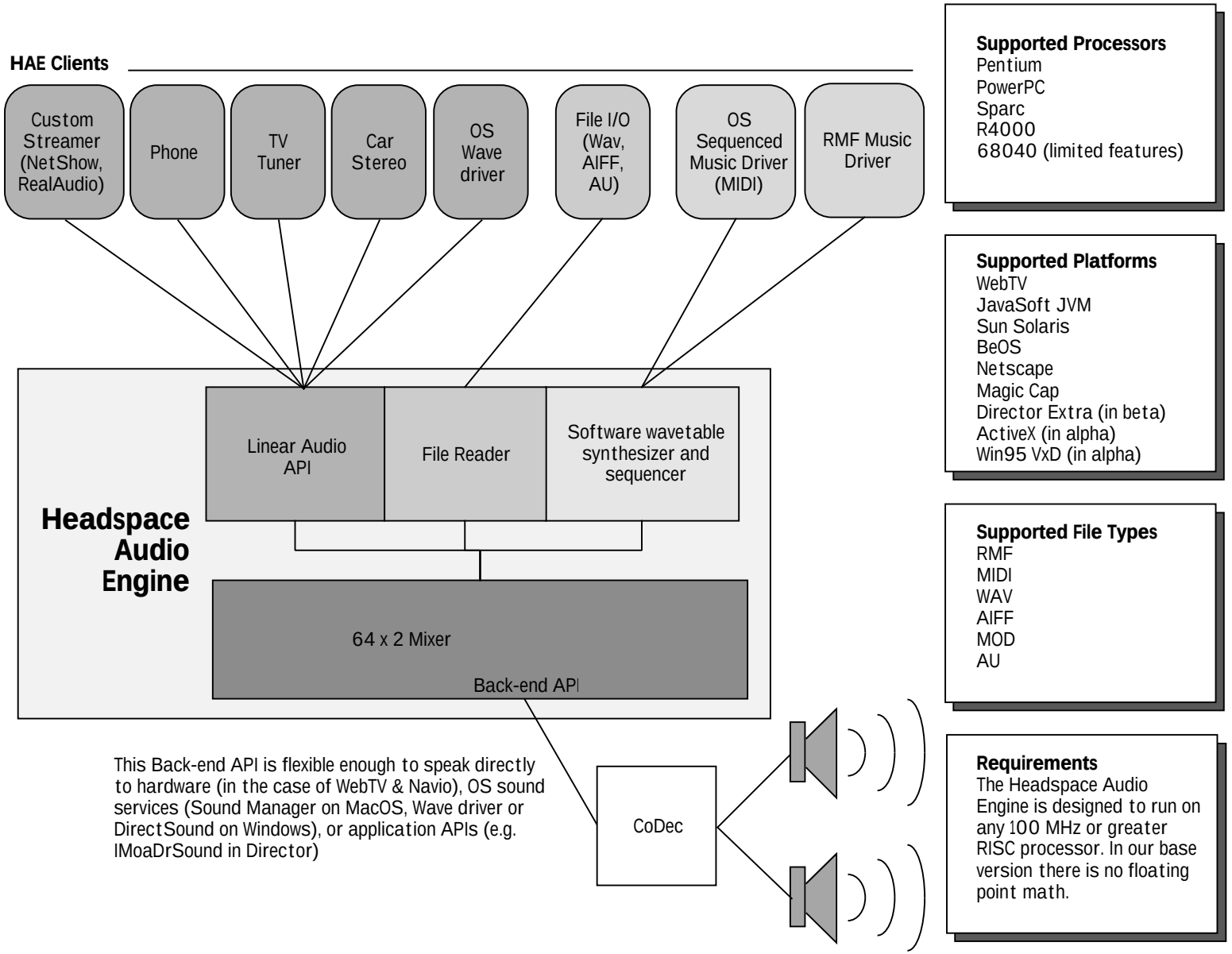
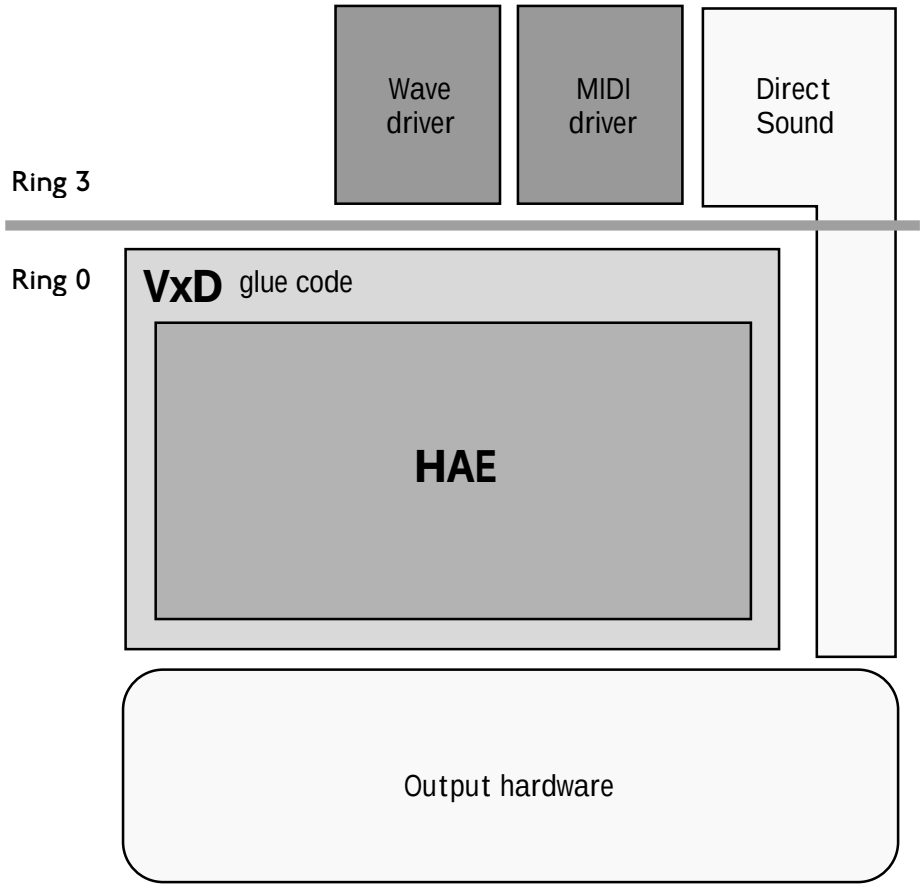

HAE Clients
Custom Streamer (NetShow, RealAudio)
Phone
TV Tuner
Car Stereo
OS Wave driver
File I/O (Wav, AIFF, AU)
OS Sequenced Music Driver (MIDI)
RMF Music Driver
Linear Audio API
File Reader
Software wavetable synthesizer and sequencer
Headspace Audio Engine
64 x 2 Mixer
Back-end API
This Back-end API is flexible enough to speak directly to hardware (in the case of WebTV & Navio), OS sound services (Sound Manager on MacOS, Wave driver or DirectSound on Windows), or application APIs (e.g. IMoaDrSound in Director)
CoDec
Supported Processors
Pentium
PowerPC
Sparc
R4000
68040 (limited features)
Supported Platforms
WebTV
JavaSoft JVM
Sun Solaris
BeOS
Netscape
Magic Cap
Director Extra (in beta)
ActiveX (in alpha)
Win95 VxD (in alpha)
Supported File Types
RMF
MIDI
WAV
AIFF
MOD
AU
Requirements
The Headspace Audio Engine is designed to run on any 100 MHz or greater RISC processor. In our base version there is no floating point math.

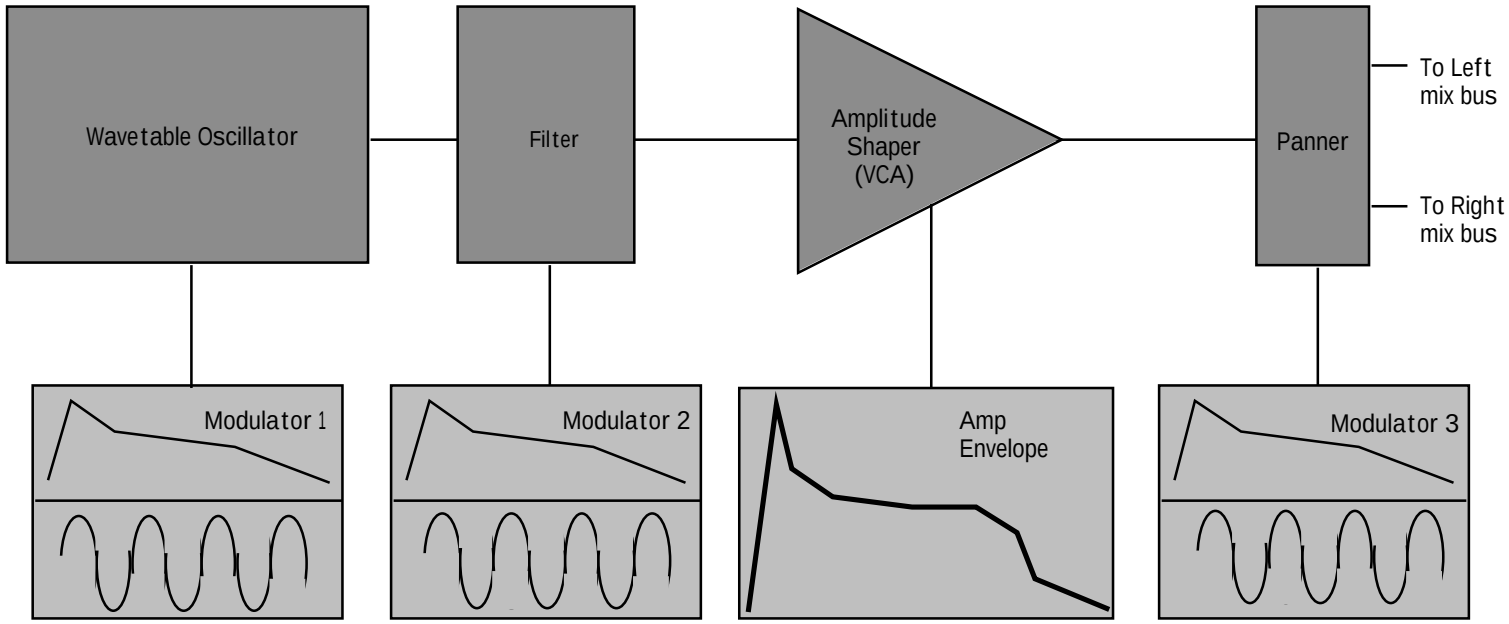
Overview



Headspace provides the version of the HAE that is embedded into the VxD glue, the VxD glue itself, the Wave and MIDI drivers, and the connection to the output hardware (or the connection to Direct Sound)

Note that the lowest levels of this world are flexible enough to allow us to:
Take the output of DirectSound and mix it into our world, then send it all off to the sound hardware.
- or -
Send the output of HAE to Direct sound, which mixes it and sends it to the sound hardware.

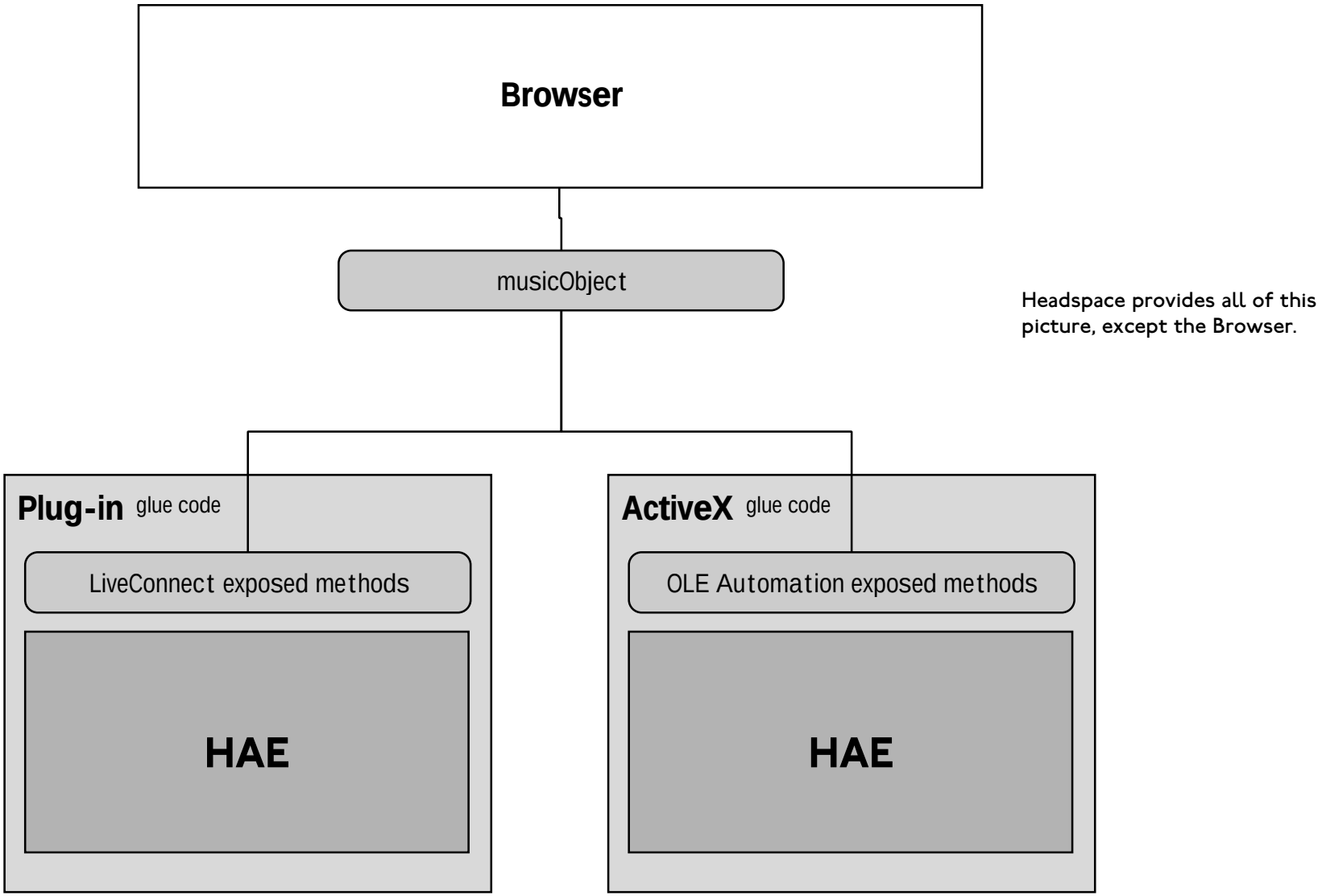
Soundcard Embedded



Each Modulator block consists of an EG (Envelope Generator) and an LFO (Low Frequency Oscillator).

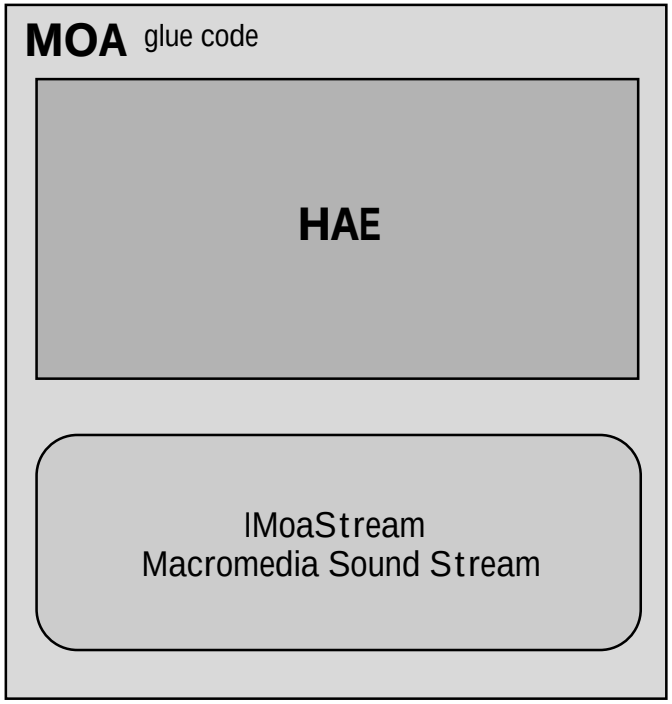
The modulator routings shown are not fixed; all three modulators could be assigned to the same destination, for example.

Synth Voice



Headspace provides all of this picture, except the Browser.

Beatnik Player



Headspace provides the version of the HAE that is embedded into MOA (Macromedia Object Archetecture) glue, the MOA glue code itself, and a sub-class of IMoaStream.

Macromedia Director Xtra

Typical code memory usage	Min	Max
ROM Footprint	[1]60K	180K
RAM Footprint (Idle)	32K	80K
Reverb buffers	[2]8K	128K

Typical Song RAM	
25 Instruments	1K
Sample Data	[3]250K
MIDI Data	30K

We have sample Banks from 400K up.

Typical other sizes are 450K, 1 Meg, & 5 Meg.

[1] No MOD Code
Not all RMF features
No Effects
Fixed output rate & depth
No file Creation code
No linear sound code
[2] Assumes our smallest Reverb code, which would add 10K to the Min. ROM code footprint
[3] Samples can be played from ROM, which brings this down to zero.

Memory Footprint