

## Abstract

Chumbi is a collection of signal processors for the Chuck programming language, enabling the creation and manipulation of BFormat Ambisonic signals. Its design philosophy prioritizes flexible tools that support traditional Ambisonic workflows while encouraging creative recontextualization of spatial audio techniques. Chumbi operates from 1st to 5th order, and its open-source code provides a foundation for users to develop higher-order processors. Chumbi's focus on modularity allows for unique approaches to designing digital instruments, enabling interfaces which explore spatiotemporal audio and data.

## Design Goals

Throughout its development, the main focus of Chumbi was to provide approachable ambisonics with plenty of room to grow. Encouraging users to learn by doing, even if done wrong.

- 1. Approachable.** Ambisonic interfaces are preconfigured in example code, allowing users to engage with the spatial domain in simple yet effective ways.
- 2. Flexible.** While preconfigured systems are provided, they can be rearranged into systems of higher complexity.
- 3. Open-Ended.** Users may fully immerse themselves in the low-level workings of ambisonic processing should they wish.

## Decoding in Chumbi

```

1 // homer simpson enters the ambisonic universe, and exits using
  Decode
2 SndBuf doh("special:doh") => Encode1 encoder => Decode1 decoder =>
  dac;
3
4 float speakerSH[4][4]; // save the spherical harmonics here
5 for(int n; n < 4; n++)
6 {
7     // math.sh calculates the spherical harmonics given a position
8     // this makes each speaker 90 degrees apart, with a 45.0 offset
9     Math.sh(1, (n * 90.0) + 45.0, 0.0) @=> speakerSH[n];
10 }
11
12 // give the decoder our spherical harmonics
13 decoder.sh(speakerSH);
14 // retrieve them
15 decoder.sh() @=> speakerSH;

```

Figure 1. Custom decoding in Chumbi, allowing for user creation of spherical harmonic weighting schemes.

## Chumbi's Contents

All of Chumbi's UGens are designed to operate in real time, for the live performance of ambisonic works and instruments.

- **EncodeN:** Encodes a virtual source using spherical coordinates (supports up to 5th order).
- **DecodeN:** A basic, customizable spherical harmonic decoder.
- **SADN:** A sampling Ambisonic decoder.
- **DBDN:** A dual-band Ambisonic decoder.
- **ABFormat / BAFormat:** Convert between first-order A-Format and B-Format signals.

Additional UGens are actively being developed for the recording of multichannel audio files as well as the warping of soundfields.

## Encoding in Chumbi

```

1 // homer simpson enters the ambisonic universe
2 SndBuf doh("special:doh") => Encode3 encoder => blackhole;
3 // go!
4 while(true)
5 {
6     // retrieve last given azimuth and rotate by a few degrees
7     encoder.azi() + 6.0 => encoder.azi;
8     1::second => now; // advance time
9 }

```

Figure 2. Encoding a virtual sound source at a specified azimuth and elevation using the EncodeN UGen.

## Custom Workflows

Chumbi offers the ability to create custom ambisonic routing, which have a variety of creative opportunities. Allowing users to create frequency-dependent encoding and decoding, spatial reverberation algorithms, and various other modular systems.

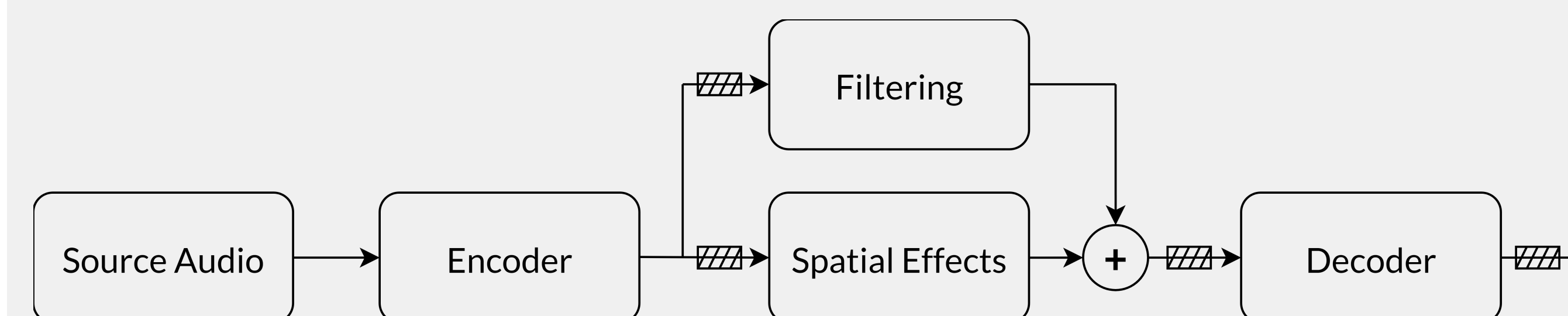


Figure 3. A flow graph of an example Chumbi workflow. Note: Lines with hash marks indicate multi-channel audio streams containing  $(N + 1)^2$  channels.

## Where to Find Chumbi

Chumbi is available for download as a collection of precompiled binaries at <https://everettmiles.com/chumbi>. Releases are available for Windows, Linux, and macOS (Intel). Chumbi is open source, forks and modifications are welcome.

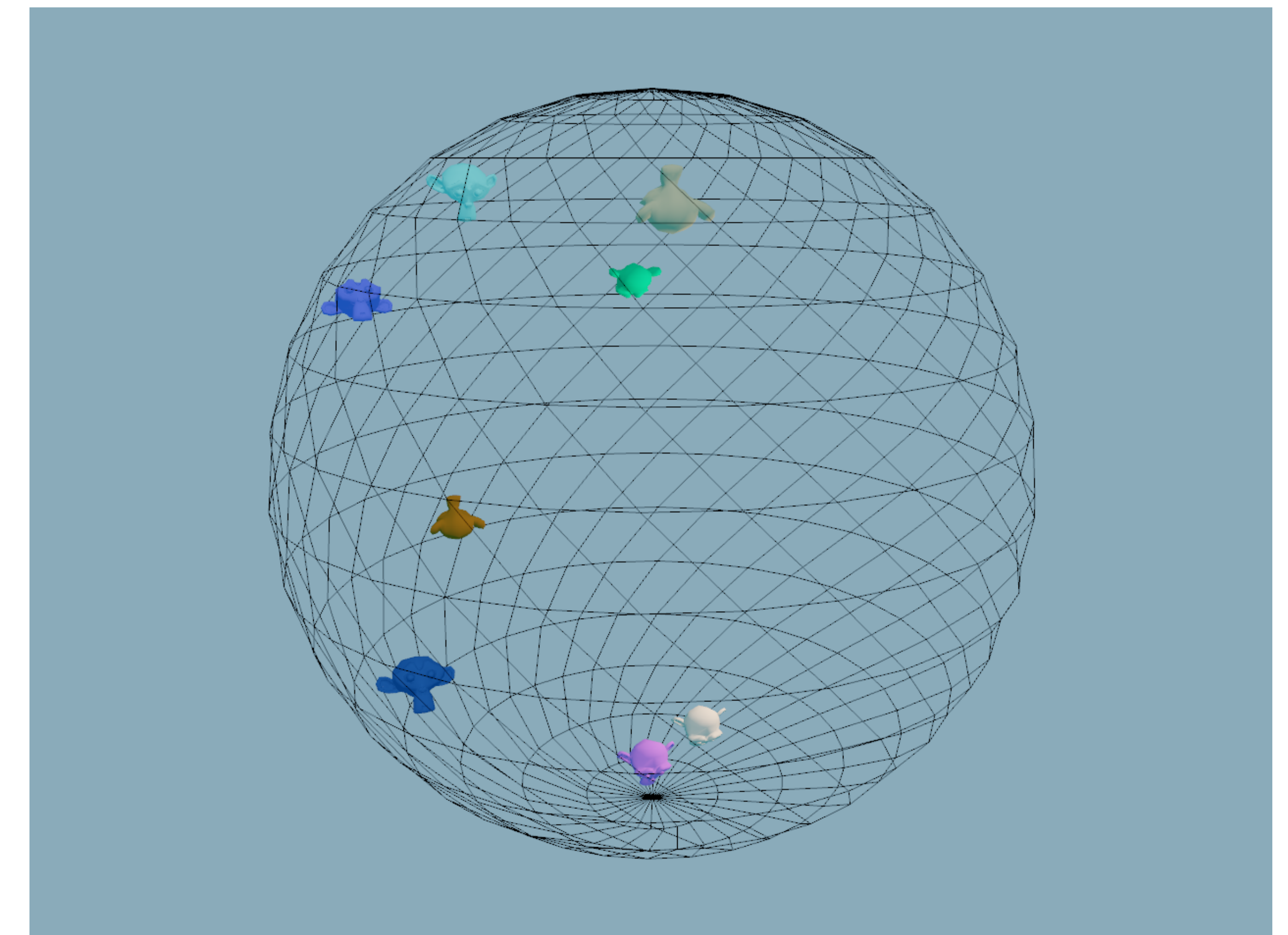


Figure 4. A spatial visualizer created in Chuck using Chumbi.

## References

- [1] Charles K. Neimog. pd-saf: Spatial audio framework for puredata. <https://github.com/EL-LOCUS-SOLUS/pd-saf>, 2025. Accessed: 2025-12-31.
- [2] IEM. lem plug-in suite, 2026.
- [3] IEM. lem pure data hoa suite, 2026.
- [4] Joseph Anderson and SuperCollider Community. Atk for supercollider. <https://github.com/ambisonictoolkit/atk-sc3>, 2025. Accessed: 2025-12-31.
- [5] C. Nachbar, F. Zotter, E. Deleflie, and A. Sontacchi. Ambix - a suggested ambisonics format. *Ambisonics Symposium 2011*, 07 2011.
- [6] C. SuperCollider. Ambisonic enlightenment, on ways to think ambisonic, a novice's guide, 2026.
- [7] P. Zinemanas, M. Rocamora, and L. Jure. Improving csound's ambisonic decoders. In *Proceedings of The Fifth International Csound Conference of 2019*, 2019.