AMBISONICS SYMPOSIUM 2009
JUNE 25–27, GRAZ

A STANDARD FOR INTERCHANGE OF AMBISONIC SIGNAL SETS
Including a file standard with metadata

Michael Chapman[1*],
Winfried Ritsch[2], Thomas Musil[2], Dhannes Zmölnig[2], Hannes Pomberger[2], Franz Zotter[2],
Alois Sontacchi[2]

[1] 01350 Culoz, France (chapman@mchapman.com)[*]
[2] IEM Ambisonics Team, Graz (ritsch@iem.at, musil@iem.at, zmoelnig@iem.at, pomberger@iem.at, zotter@iem.at, sontacchi@iem.at)
[*] *chairman and author for correspondence*

Abstract: *Over the past few years there has been growing awareness of the need for an agreed format for ambisonic files and for the interchange of other ambisonic signal sets.*
Here we propose a standard that is both simple and intended to be future proof.
The proposal is the outcome of many months of discussion, on the Web and by email, and of physical meetings: Parma (April 2009 at LAC2009, courtesy of Fons Adriaensen) and Graz (June 2009 at the IEM Ambisonics Symposium, courtesy of Professor Ritsch and colleagues) where this final draft is due to be agreed.
The proposal is for N3D normalisation, Ambisonic Channel Number sequence and the use of a UUID chunk in the Apple Core Audio Format *(.caf) for metadata about included channels.*
The file format may also be used for non—B-format material (A–, C–, etc. format), for position files, etc. The metadata is extensible to allow for future possible uses.

Key words: file format, signal set format, interchange, standard, ambisonics.

1 INTRODUCTION

There is an obvious need for a standard so that ambisonic workers can interchange material, so they can publish files or streams and so that software (and equipment) can be created that is usable to all.

The current *de facto* standard, FuMa (see below), is no longer tenable as a universal format as higher order materials become more common. It also lacks a method for transmission of data about the audio data (‘metadata’), meaning that signal sets will often have to have this data conveyed by means external to the audio file.

We very briefly review the relevant history of ambisonics. Then on the basis of many discussions and consultations (not least those detailed on the IEM Website [1]) propose a ‘futureproof’ format.

2 HISTORY

Ambisonics uses a spherical harmonic representation of a soundfield to allow a full sphere (or full circle) soundfield to be recorded or synthesised and then played back.

Whilst spherical harmonics have a long history (1784 for the sphere (Laplace) and the preceding year for the circle (Legendre)) and had been used in acoustics, it was only late in the last century that Michael Gerzon proposed their use for audio, with the name ‘ambisonics’ first published in 1971. His and Peter Fellgett’s work pioneered this field.[3]

[3] Michael Anthony Gerzon 1945—1996
Peter Berners Fellgett 1921—2008

Initially, as can be imagined, ambisonics used the most basic set of data feeding the most basic practicable rig of loudspeakers. That is three channels decoded to four speakers.

What we argue, here, is the most logical normalisation scheme for the relative amplitudes of the channels was used. That is full normalisation. (A full set of spherical harmonics is such that the sum of the squares of the individual values in each degree is equal to the number of values in that degree. See Daniel [2] for a precise definition of normalisation schemes, their nomenclature and interconversion.) For two–dimensional usage (pantophony) that meant N2D.[4]

Use cases obviously grew, though. For higher orders of pantophony N2D remained the logical choice. For periphony (three–dimensional) uses then a new component (termed Z then, and here channel 2) was added, *but* normalisation stayed with the criteria for the then predominant two–dimensional usage.

Users who wished to extend ambisonics were then confronted with the problems we now face: The choice of relative amplitudes for the channels and a sequence for them.

The work of Bamford & Vanderkooy [4] had already extended pantophony to second order, adding channels known as U and V. Furse and Malham [6] proposed a system, now called FuMa (or FMH), incorporating that extension, and extending periphonic ambisonics beyond first order. This has enjoyed wide popularity and is an undoubted success.

The original channel names W and XYZ were kept for the zero and first degree elements and the letter codes U and V used for the pantophonic components of second degree with R to T added for the other second degree components. Letter codes K to Q were added for third degree. Malham, himself, commented in “Higher order Ambisonic systems” (at p. 3, see [6]) that whilst the letter codes could be extended to fourth order:

> I do not, in general, think it worth continuing the use of the letter based nomenclature for channel names above the third order, although the English alphabet would actually accommodate the nine channels of fourth order, . . . ”.

A practically unimportant, change was also made in that W,X,Y,Z had their amplitudes reduced by the square root of two.[5] This meant that for a plane wave all channels now had a maximum possible value of 1, except for W which had a possible maximum of 0.7071. This approach (MaxN in Daniel’s terminology, except for W) was extended to the ‘new’ channels by the use not of normalisation but the use of ‘weightings’. This scheme readily shows up problems in software/hardware manipulation of test signals, but is tortuous to work with as the numbers of channels increase. Also whilst useful for plane wave test signals it does not use the channels effectively for a natural soundfield.

The success of the FuMa scheme was great. It was though exceeded by the success of ambisonics which it had itself engendered. Users were left with a foreseeable exhaustion of channel names, and the relative amplitudes of those channels becoming increasingly unwieldy to work with.

3 THE FORMAT

(*A more formal and complete definition occurs in the Annex.*)

For normalisation we return to Gerzon’s original use of full normalisation, *but* use three-dimensional full normalisation (N3D). This gives the best possible use of the channels for a natural soundfield.[6] It is also, fortuitously, considerably easier to understand and to implement.

There are undoubtedly good arguments for having SN3D [2] as well as N3D. The ease of interconvertibility, the not great difference in peak amplitudes[7] and the ease of compressing headroom do not strongly argue for offering both. Even though SN3D is undoubtedly easier to use in some situations, conversion is easy. The overwhelming argument is that of simplicity and of only having one normalisation in this file standard.

The FuMa first–degree channel sequence of XYZ is decep-

[4] Interestingly there is a presumption that somehow SN3D is ‘normal’ (in the sense of being the norm) in ambisonic literature referring to spherical harmonics. The difference between SN3D and N2D leading to apologetic comments. Benjamin *et al.* [3] as recently as 2006, though approaching this more objectively, commented (footnote on their page 3):

> The additional scaling factor of $\frac{\sqrt{2}}{2}$ in the W component is a historical artifact. It was added to improve the utilization of the dynamic range of recording media, based on the observation that the typical signal levels in the W channel are several dB higher than in X, Y or Z.

(W is B_0, X,Y and Z the first degree components.) Whilst we would disagree that this is likely to be an accident of history, we do agree on the practical merits of full two-dimensional normalisation for what were then predominantly pantophonic signal sets.

[5] The authors are grateful to Chris Travis (citing [5] and patent US5757927, of 1998) for pointing out this easily overlooked piece of history to one of their number. Page 4 of [5] reads: “W [has] . . . [a gain] equal to one, X . . . with a gain $2^{\frac{1}{2}} \cos\theta, \ldots$”.

[6] In January 2009, Fons Adriaensen commented:

> The basic fact that for a random (which means expected isotropic) distributions N3D is optimal follows directly from basic theory. That it holds in practice even if you take real-life (max-rE) decoding gains into account can be verified experimentally, as I’ve done now at least three times, using different code each time.

(On the SURSOUND mailing list. Reproduced here with the author’s kind permission.)

See also figure 1.

[7] A four bit difference (a factor of sixteen) occurs at about degree 127!

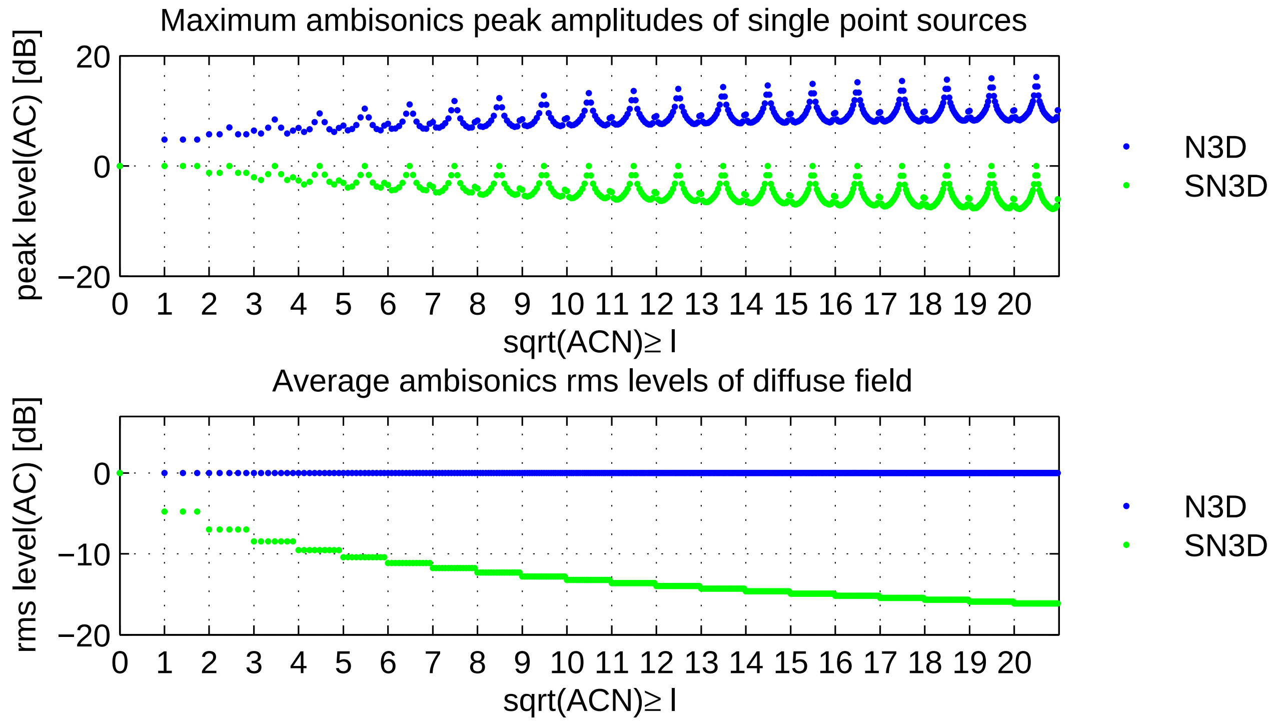

Figure 1: The upper diagram shows the maximum accessible peak dBFS-level of the ambisonic channels (referred to by ACN) when encoding with N3D and with SN3D normalisation schemes. These maximal levels are obtained by storing the maximum level for all angular encoding angles.The lower diagram depicts the average dBrms-levels of spatially diffuse sounds. The latter is regarded as the asymptotic limit for equally distributed, very dense sound scenes encoded in ambisonics.

tively attractive, as the related spherical harmonics align along those axes. There is no such simple pattern for higher degrees. We have therefore given each channel a unique integer (≥ 0) and placed those channels present in their numerical sequence. That integer (the *ambisonic channel number*[8], ACN) is derived from the coefficients of the related spherical harmonic: $l(l+1)+m$.

A fuller technical discussion of the above occurs in a paper presented to this Symposium [7].

For file usage these channels are placed in an Apple “Core Audio Format” [8] file (a .caf file, with that suffix).

The file must comply with the CAF specification.

It must also have a UUID chunk before the audio data.

That chunk must have the UUID “5dc3f270c2d24293858 e64da38090bea”, and contain metadata indicating (for B-format) which channels are present. The metadata may also specify that the content is of a format other than B–format, see below.

There may also be additional UUID chunks after the audio data. These should be referenced from the main metadata chunk and are intended for programme notes, artwork, etc.

3.1. Metadata

For B-format (the normal format for the interchange of ambisonic material) the metadata specifies that

- the file is B-format
- the channels present

The latter is achieved by using (H,V,P) notation (see Travis’s paper to this symposium). (H,V,P) allows all pure pantophonic, pure periphonic and ‘classical’ mixed order signal sets to be represented, *as well* as allowing for more adventurous mixed order sets. By ‘classical’ we mean sets that can be represented in Malham Notation, with lower degree(s) periphonic and higher degree(s) pantophonic.

It should be noted that Franz Zotter and Hannes Pomberger are working on a matrix system for representing signal sets for partial sphere playback rigs. It seems likely that this will require very specific software for decoding. Such an extension, once published, can easily be included in the metadata

[8] Therefore channels 0 to 15 correspond respectively to W YZX VTRSU QOMKLNP.

specification. At the moment there seems little value in replacing (H,V,P) with it, rather to offer it as an alternative to it.

The need for other formats –other than B-format– is debatable, but on balance the authors have concluded that if such files exist they would better exist with proper descriptive metadata and thus have been included. Certainly they have some uses in education (students can work with an A-format file rather than needing a live microphone feed, for example) and other very specific cases (see the 'position' file described below). The required metadata for these is described in the metadata specification DTD.

The metadata is included as XML. It has been suggested that a simpler description based on OSC or JSON would be 'cleaner'. 'Patches' for either of these, or other, representations are welcomed. The DTD for the metadata can be found at `http://ambisonics.ch/dtd/ambisonic.dtd`. At the time of writing this is version `ambisonic-0.06.dtd`, the metadata should give a URL to the specific version, not the generic URL (as in the previous sentence –this returns the latest version).

4 IMPLEMENTATIONS

Currently AMBISUITE [9] produces files in this format. It also allows users to interrogate a file and be informed of the metadata present in the file. It provides for basic manipulations (rotation, dominance) on such files. It converts legacy formats (i.e. .amb) to this format. It will be able to 'mend' files that have lost their metadata.

AMBDEC [10] will decode files in this format (upto second order), the author has indicated he will write a new decoder for higher orders with inputs labelled in the ACN notation.

The author of TETRAPROC [11] has also offered to create a version that decodes feeds from a first order microphone to the new format.

This format is, therefore, usable now.

5 COMPRESSION

The authors know of no way of compressing PCM-encoded CAF files. However WAVPACK [12] compresses (in lossless, lossy and 'hybrid' modes) PCM audio data for WAV files with arbitrary channel counts.

David Bryant (WAVPACK's developer) has been approached by the authors and is working on extending its capabilities to the compression of CAF files. This will hopefully be available this summer. Limitations in the underlying PCM compression libraries relating to channels ($\leq$ 16) and file size ($\leq$ 4GB) are already being worked on and would automatically benefit CAF compression as they are introduced to the code. (Giving $\leq$ 65,536 channels and the general limit on CAF file size. The latter is described by Apple as "unlimited", in reality it appears to be $2^{64} - 1$ bytes (over ten million TB).) The limit on definition ($\leq$ 32-bits) is unlikely to be removed, but is equally unlikely to be of relevance for ambisonic usage.

Obviously the use, or not, of compression is the user's choice. Applications should accept and should offer to produce compressed files whenever possible.

The WAVPACK libraries can be used to quickly access the metadata of a compressed file without decompressing it, so an application can easily *(i)* confirm it is a file declaring itself as complying with this standard, and, *(ii)* return the contents of the metadata for users. Because the metadata is not itself compressed and is stored early in the file, it would even be possible (though not preferable) to search for and access the metadata directly.

6 EXTENSIBILITY

Placing the metadata in a chunk of arbitrary length makes it very easy to extend the metadata.

One example that occurred during development has been the addition of 'position' files. These allow test files (of the "Up Left Front, Up Front, ..." variety) to be distributed as four channel files (mono, u_x, u_y, u_z), which can then be 'inflated' by the user to a normal file of any ambisonic order.

The ease of extensibility needs using with care. It also requires a guardian of the metadata standard. We propose that the metadata standard be maintained by the Geneva based Ambisonics Association which already hosts the DTD on its webserver (see above).

7 DISCUSSION

There will undoubtedly be criticisms of this proposal. The two main ones would seem to be:

A lack of backwards compatibility: Frankly we cannot see this as a problem. Files can be converted. A converter from AMB[9] to this format is already published. We refute arguments that FuMa material above first order does not exist (e.g. Hofmann's compositions) or that software that handles such material does not already exist (e.g. AMBDEC). The only new file format that is going to be truly backwards compatible is going to be the old file format(!).

[9] The current main method of exchanging ambisonic data: using a WAV type file with FuMa data.

The 'problem' of files without metadata: These will always be a problem. In the authors' opinion it is better that this is clearly so, then perhaps users may stop publishing them(!). Some of the workarounds being proposed are indeed elegant. They might be parodied as requesting users to create image files with height and width both prime numbers of pixels, so that software can derive the correct aspect ratio. One proposal is to fix the positions of the first six channels (this goes against backwards compatibility) and then use software to determine what the other channels are (perhaps akin to trying to match rows of pixels to determine row length). We know of no proposal as to how this should be done (and indeed have some doubts if it can be done for synthesised compositions), nor of anyone working on developing such tools. We do not deny the elegance of the proposal.

8 ACKNOWLEDGEMENTS

David Bryant freely gave of his time to advise us on file compression and has been most accommodating to our requests for enhancements to WavPack.

The authors would also like to thank the many members of the ambisonic community who participated in debates on this subject, in various fora. Their contributions, both for and against various points, have been invaluable.

REFERENCES

[1] http://ambisonics.iem.at/xchange/format
http://ambisonics.iem.at/xchange/meeting08

[2] see pages 155–157, and especially table 3.2, of:
Daniel, Jérôme. *Représentation de champs acoustiques, application à la transmission et à la reproduction de scènes sonores complexes dans un contexte multimédia*. Thèse de doctorat de l'Université Paris 6, 31 juillet 2001.
(Also at: http://gyronymo.free.fr/audio3D/downloads/These.pdf)

[3] Benjamin, E.M., R. Lee & A.J. Heller. *Localization in Horizontal-Only Ambisonic Systems*. Revised and corrected version (October 8, 2006) of paper to AES 121, October 2006, thirteen pages.

[4] The following:
Jeffery S. Bamford & John Vanderkooy. "Ambisonic Sound for Us". Audio Engineering Society, 92nd Convention, preprint 3345, March 1992.
is cited by Daniel and was informally drawn to the attention of one author. Unfortunately a copy was not obtained prior to this submission.

[5] Gerzon, Michael A. & Geoffrey J. Barton. "Ambisonic Decoders for HDTV". Audio Engineering Society, 99th Convention, preprint 4138, October 1995.

[6] There appears to no be definitive publication (as so often in ambisonics) for FuMa. At http://www.york.ac.uk/inst/mustech/3d_audio/secondor.html Dave Malham reports that the Furse–Malham proposal was originally published in 1999 (Malham, D.G. 'Higher order Ambisonic systems for the spatialisation of sound' *Proceedings, ICMC99*, Beijing, October 1999), later though the weightings were adjusted and then the channel sequence was changed ("third order components have ... been reordered"). The current signal set is in "Higher order Ambisonic systems" (at http://www.york.ac.uk/inst/mustech/3d_audio/higher_order_ambisonics.pdf) of 2003 abstracted from Malham's MPhil thesis "Space in Music — Music in Space" of the same year.

[7] Chapman, Michael & Philip Cotterell, 2009. "Towards a comprehensive account of valid ambisonic transformations". Paper submitted to the First Ambisonics Symposium, IEM. Institute of Electronic Music and Acoustics, Graz, Austria, June 25-27, 2009.

[8] *Apple Core Audio Format Specification 1.0*, Apple Inc. (1 Infinite Loop, Cupertino, CA 95014, USA). 2006-03-08. 62pp.
(Also at http://developer.apple.com/documentation/MusicAudio/Reference/CAFSpec/CAFSpec.pdf)

[9] http://ambisuite.sourceforge.net/
Chapman, Michael, 2009. "Ambisuite/Ambman: a utility for transforming Ambisonic files", *Proceedings Linux Audio Conference 2009*, pages 53–59. Parma, Instituzione Casa della Musica, 196pp. (Also at http://lad.linuxaudio.org/events/2009_cdm/Thursday/06_Chapman/06.pdf)

[10] Adriaensen, Fons, 2008. *AmbDec - 0.2.4 User Manual*, Wed 15 Oct 2008, 10 pp.
http://www.kokkinizita.net/linuxaudio/downloads/ambdec-manual.pdf

[11] Adriaensen, Fons, 2007. "A Tetrahedral Microphone Processor for Ambisonic Recording", 6 pp. *Proceedings Linux Audio Conference 2007*.
http://www.kokkinizita.net/papers/tetraproc.pdf

[12] WavPack and WvUnpack
http://www.wavpack.com/.

The ANNEX occurs on the next page.

Specification

A file in this format:

- **is a CAF file complying with the Apple "Core Audio Format" specification**
- **it has the file suffix `.caf`**
- **must not have a Channel Layout chunk ('chan').**
- **must have a UUID chunk before the audio data**
  - **that chunk must have the UUID "5dc3f270c2d24293858e64da38090bea"**
  - **that chunk must contain the metadata detailed below**
- **may have additional UUID chunks after the audio data**
  - **if present this/these should be referenced from the main metadata chunk**

The included audio data must:

- **be encoded in PCM (integer or floating point) of up to 32-bits (whilst 64-bit is not 'illegal' it will not be supported by compression, and probably by few if any applications)**
- **be N3D normalisation**
- **have the channels in ACN order**

The metadata must be well–formed XML that validates against the DTD at `http://ambisonics.ch/dtds/ambisonic.dtd`, and must state the version number of the DTD used.

Compliant applications must produce files that comply with the above if writing files. Compliant applications that produce streams must comply with audio data requirements above. Applications designed to produce broadcast streams must provide commencing metadata and should repeat this throughout broadcast as required by the streaming standard.
Compliant applications that read files, or accept streamed files, must accept any file in this format. They must then, either:

- **return a message stating their inability to work with the data in the file —this message should be verbose enough for the user to understand the nature of the file/stream and the reason for the inability; or,**
- **handle the data correctly.**

Wherever practicable, compliant applications that read and/or write files should accept/generate WAVPACK compressed files. For writing this may be as a user specified option and may extend to options between uncompressed, lossless, lossy and 'hybrid' compression.

June 2009