Fraunhofer IIS

WHITE PAPER

xHE-AAC – ADAPTIVE BIT RATE AUDIO CODEC FOR MUSIC AND VIDEO STREAMING

INTRODUCTION

This document intends to give implementation and integration advice when deploying Fraunhofer's new codec xHE-AAC, based on the most recent member of the MPEG AAC audio codec family, Extended High-Efficiency AAC and the latest MPEG loudness control technology, MPEG-D DRC. The authors describe technical details bearing in mind specific use cases and deployment scenarios, which are described at the beginning of the document.

The technical specifics are accompanied by general background on audio coding, with a view to establishing an understanding of the signal processing fundamentals found in modern audio codecs and the way they exploit human psycho-acoustic phenomena.

Fraunhofer Institute for
Integrated Circuits IIS

Management of the institute
Prof. Dr.-Ing. Albert Heuberger
(executive)
Dr.-Ing. Bernhard Grill
Am Wolfsmantel 33
91058 Erlangen
www.iis.fraunhofer.de

Contact
Matthias Rose
Phone +49 9131 776-6175
matthias.rose@iis.fraunhofer.de

Contact USA
Fraunhofer USA, Inc.
Digital Media Technologies*
Phone +1 408 573 9900
codecs@dmt.fraunhofer.org

Contact China
Toni Fiedler
Phone +86 138 1165 4675
china@iis.fraunhofer.de

Contact Japan
Fahim Nawabi
Phone: +81 90-4077-7609
fahim.nawabi@iis.fraunhofer.de

Contact Korea
Youngju Ju
Phone: +82 2 948 1291
youngju.ju@iis-extern.fraunhofer.de

* Fraunhofer USA Digital Media Technologies, a division of Fraunhofer USA, Inc., promotes and supports the products of Fraunhofer IIS in the U. S.

BENEFITS OF xHE-AAC

xHE-AAC is a mandatory audio codec in Google's Android Pie mobile operating system based on Fraunhofer's FDK software. The codec combines maximum coding efficiency, seamless bit rate switching and built-in MPEG-D DRC Loudness and Dynamic Range Control, making it the ideal solution for audio and video streaming services as well as digital radio broadcasting.

Inherently designed for adaptive streaming, xHE-AAC provides reliability for streaming services even under the most challenging network conditions. xHE-AAC is the only perceptual audio codec that covers the entire bit rate spectrum – starting as low as 6 kbit/s for mono and 12 kbit/s for stereo services. Thus xHE-AAC streaming apps and streaming radio players may switch to very low bit rate streams and offer a continuous playback even while the network is congested. Once more bandwidth becomes available on the network again, the xHE-AAC client application can request a higher bit rate version and seamlessly switch over the full range of bit rates. The saving in audio bit rates due to the improved coding efficiency can be used to improve video quality. In addition, MPEG-D DRC provides mandatory loudness control for xHE-AAC to play back content at a consistent volume and offers new dynamic range control profiles for listening in noisy environments.

Developers will be able to use the new features immediately in Android Pie, which offers extensions to the existing FDK APIs [OMX.google.aac.decoder]. The new technologies have been included in the existing, successful AAC patent licensing program administered by VIA Licensing at no additional cost. Professional xHE-AAC encoding and decoding software solutions are available from Fraunhofer today.

TECHNOLOGY ESSENTIALS

The "AAC family"

The MPEG-4 standards suite

Historically the MPEG audio codecs have evolved over the past decades, with MP3 being the the first and most prominent in a successful line of codecs. After the standardization of Layer 1, 2, and 3 (aka MP3) in MPEG-1 and MPEG-2, MPEG set out to create a new audio codec, designed from scratch, that would resolve some of the more fundamental shortcomings of MP3 and would drastically improve coding efficiency: Advanced Audio Coding (AAC). AAC became part of a much larger suite of standards called "MPEG-4", which contained everything required for very complex compositions of media including video coding standards, in particular Advanced Video Coding, AVC (a.k.a. H.264), systems components defining transport and presentation of media, and many more parts. The worldwide proliferation and hence the significance of the various MPEG-4 parts varies strongly, though certain parts, including AAC, AVC, the ISO Base Media File Format (ISOBMFF) and others, certainly represent the most commonly used media codec and transport ecosystem in the world.

MPEG-4 Audio Object Types

The audio part, Part 3, of MPEG-4 [MP4A] is itself highly complex and again defines a whole suite of audio codecs, all of which can be addressed and employed within the MPEG-4 ecosystem. In order to be able to identify or signal the type of audio codec used, MPEG-4 defines what is known as an audio object type (AOT), which enumerates all specified audio codecs and technologies. The most important audio object types that will be the subject of this document are

- AOT 2: AAC Low Complexity (AAC-LC)
- AOT 5: Spectral Band Replication (SBR)
- AOT 29: Parametric Stereo (PS)
- AOT 42: Unified Speech and Audio Coding (USAC) [USAC].

Of these four AOTs, SBR and PS are special in that they cannot be used by themselves; they require a core audio codec with which they interoperate. Though this sounds complicated, the good news is SBR is effectively only ever used in combination with AAC-LC; and PS hooks onto SBR.

With respect to AOTs 2, 5, and 29, MPEG-4 defines the following combinations of AOTs:

- 2 only (AAC-LC)
- 2 + 5 (AAC-LC + SBR)
- 2 + 5 + 29 (AAC-LC + SBR + PS)

Since PS was only ever combined with SBR and AAC-LC, "AOT 29" became synonymous with what is actually the combination of AOTs 2 + 5 + 29. Similarly, "AOT 5" is used synonymously for the combination of AOTs 2 + 5.

USAC deviates from the AOT scheme by inherently defining its own enhanced version of SBR and PS, thus there is no further need to combine it with other AOTs at the signaling level even though it does provide the same kind of (and more efficient) algorithmic features.

Profiles

In order to keep the complexity of the standard manageable and in order to provide sensible, realistically deployable subsets of technology, MPEG defines so-called profiles, which lay out detailed interoperability constraints for encoder and decoder implementers. Most notably the profiles restrict the use of AOTs, but also other aspects that influence computational complexity at the player side, such as channel configurations (number of audio channels), sampling rate etc.

In short, the abovementioned combinations of AOTs each have their profile counterpart. AOT 2 by itself is captured in the "AAC Profile." AOT 2 in combination with AOT 5 is defined in the "High Efficiency AAC Profile" (HE-AAC or HE-AACv1). The combination of the AOTs 2, 5 and 29 is covered by the "High Efficiency AAC v2 Profile" (HE-AACv2).

A decoder that can further decode AOT 42 is then defined in the "Extended High Efficiency AAC Profile."

The idea behind the cascaded hierarchical structure is to ensure that all new decoders are also capable of decoding audio encoded using the older, less advanced audio object types (backward compatibility).

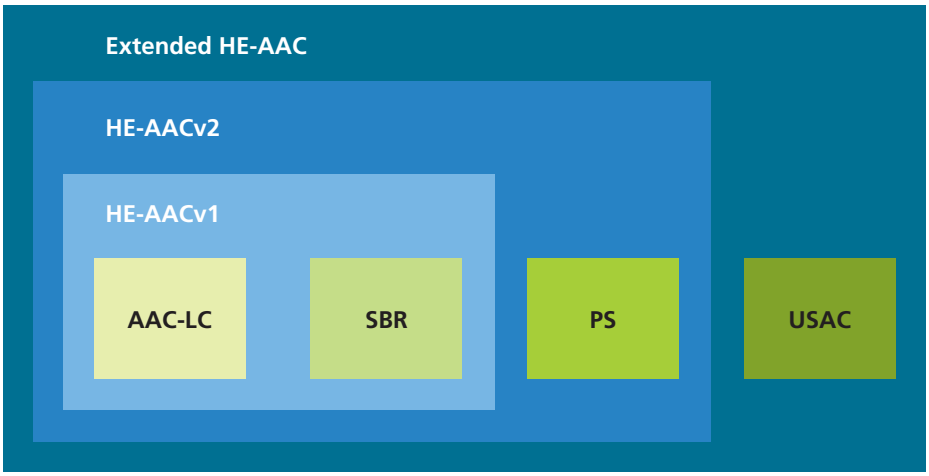

Figure 1: Profiles and object types of the AAC codec family

To complete the picture, the USAC standard further defines a "Baseline USAC" profile, which covers only AOT 42. The following table summarizes the relationship between AOTs and profiles.

Formal name	AOTs	Commonly known as
AAC Profile	2	AAC-LC
High Efficiency AAC Profile	2, 5	HE-AAC
High Efficiency AAC v2 Profile	2, 5, 29	HE-AACv2
Baseline USAC	42	
Extended High Efficiency AAC Profile[1])	2, 5, 29, 42	xHE-AAC (incl. MPEG-D DRC)

[1] superset of HE-AACv2 and Baseline USAC

DYNAMIC RANGE AND LOUDNESS CONTROL ESSENTIALS

INTRODUCTION

Studies have shown that consumers prefer listening to audio and multimedia content presented at similar loudness levels. Consumers do enjoy variations in loudness within an item for artistic effect, as long as there is not a jarring non-creative transition to another item. Audio and multimedia content is enjoyed using various devices ranging from smartphones to home theaters. Mobile devices are used in acoustic environments that may severely impact the playback audio quality. Moreover, a multitude of different content is available and accessed instantaneously at random. Loudness variations that may be tasteful and appreciated in a quiet listening environment may be too broad for listening in a noisy environment such as in a car or public place.

To be more specific, here are a few examples to illustrate some common problems that may result from the scenarios described above:

- The user needs to adjust the volume because the loudness of different items is not consistent.
- The intelligibility of movie dialog is adversely affected in soft parts due to a noisy listening environment.
- The level of loud parts of a movie is annoyingly high when soft parts are just loud enough; or soft parts are inaudible when loud parts are at a reasonable level.
- The dynamic range of an item is too large for the employed playback device (e.g. low-quality loudspeakers) or for the desired playback level.
- The audio signal clips after a downmix.
- Existing dynamic range compression of the playback device is not aware of the content characteristics and may degrade the audio quality beyond expectation.

Given this scenario, dynamic range and loudness control are important tools with which to enhance perceived audio quality and consumer satisfaction.

Technical solutions to loudness handling have evolved during the past decades as source-side or playback-side techniques. In source-side solutions, signal processing is employed before content distribution to produce a desired loudness level (e.g. -24 LKFS) and dynamic range at the playback side.

Playback-side solutions perform adjustment of the loudness and dynamic range in the playback device. This allows a single bitstream to be used for different playback devices (e.g. home theater, TV set, mobile device) and different listening environments (e.g. silent living room, noisy public transport).

xHE-AAC includes a built-in playback-side solution for dynamic range and loudness control specified by the MPEG-D DRC standard.

The MPEG-D DRC standard

The MPEG audio standard for Dynamic Range Control (MPEG-D DRC) [DRC] defines a flexible metadata format to support comprehensive dynamic range and loudness control. It addresses a wide range of content delivery use cases including media streaming and broadcast applications. In the case of xHE-AAC, MPEG-D DRC metadata is attached to the encoded audio content and can be applied during playback in a flexible way to enhance the user experience in various playback scenarios. The integrated loudness control can for example be used to meet regulatory requirements or for loudness normalization across different audio applications and content types.

MPEG-D DRC Profiles

In the same way as for the AAC family of codecs, MPEG has defined two MPEG-D DRC profiles to keep the complexity of the standard manageable, to provide sensible, realistically deployable subsets of technology and to lay out detailed interoperability constraints for encoder and decoder implementers.

The following two profiles are defined, which are both supported by xHE-AAC:

- Loudness Control: Provides all features necessary for loudness control only.
- Dynamic Range Control: Provides all features for loudness control and DRC.

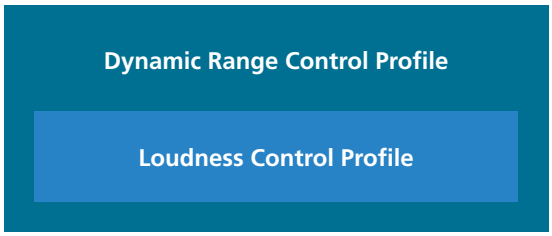

Figure 2: MPEG-D DRC Profile Hierarchy

In addition to the minimum requirements in terms of decoder tools, both MPEG-D DRC profiles define minimum requirements for the presence of what are known as basic metadata sets in a profile compliant bitstream. These mandatory metadata sets include the measured content loudness and dynamic range control instructions for different listening conditions. Each xHE-AAC bitstream must contain these basic metadata sets conforming to the Loudness Control Profile or to the Dynamic Range Control Profile. It is up to a service provider to include additional metadata sets that provide flexibility in the trade-off between functionality and bit rate.

Content Loudness

For generation of basic metadata sets, it is required to provide at least the "Content Loudness" to the xHE-AAC encoder. The content loudness represents a long-term integrated measurement of the loudness level of a file or an audio stream that is fed to the xHE-AAC encoder. The content loudness serves as the essential parameter for generating dynamic range control metadata at the encoder and for content normalization at the decoder, dependent on the specific receiver type and listening scenario.

There are two content loudness paradigms that are relevant in practice:

- "Program Loudness": The content loudness represents the long-term integrated loudness of the full program mix of a file or audio stream.
- "Anchor Loudness": The content loudness represents the loudness level of an anchor element that is part of the full program mix.

In the latter case, dialog is usually used as the anchor element, since it closely represents the overall or average subjective loudness of a piece of content. If no anchor element is present in a mix (e.g. for music content), the program loudness is used as anchor loudness.

Dependent on the specific xHE-AAC use case, either program or anchor loudness might be preferred or required to be applied based on applicable loudness regulations.

Content Loudness Handling for xHE-AAC Encoding
For file encoding, there are three possible workflows for content loudness handling:

The "Pre-Normalization Workflow" allows for easy and reliable operation if all audio content is pre-normalized to the same content loudness before entering the xHE-AAC encoder. This workflow is recommended if a workflow with varying content loudness or a two-pass loudness measurement workflow as explained below cannot be realized. In this case, the xHE-AAC encoder is operated with a fixed content loudness setting (e.g. -24 LKFS).

The "Agile Loudness Workflow" allows for the largest flexibility in terms of content dynamics and loudness variations. This workflow is recommended if the original content dynamics should be preserved. In this case the xHE-AAC encoder is operated with a variable content loudness setting, which is dependent on the individual WAV file. Note that the "Pre-Normalization Workflow" might require destructive dynamics pre-processing if high dynamic range content is pre-normalized to higher levels without sufficient signal headroom.

If content pre-normalization or reliable agile loudness cannot be guaranteed under all circumstances, the "Loudness Measurement Workflow" is strongly recommended. In this workflow, the content loudness of a WAV file is measured right before encoding and the correct content loudness is automatically passed to the xHE-AAC encoder. Note that this two-pass process can be optionally combined with a pre-normalization stage in between loudness measurement and the xHE-AAC encoder.

For the encoding of live sources, there are two possible workflows for content loudness handling:

The "Automatic Leveling Workflow" allows for easy and reliable operation if live sources are pre-conditioned by an automatic loudness leveling process. In this case, the xHE-AAC encoder is operated with a fixed content loudness setting, similar to the "Pre-Normalization Workflow" for file encoding, where the content loudness setting is aligned to the target loudness setting of the leveling process. This workflow is recommended for live sources for which manual loudness leveling and consistent content loudness cannot be guaranteed under all circumstances.

The "Manual Leveling Workflow" allows for the largest flexibility in terms of content dynamics and loudness variations for the encoding of live sources. In this case, the xHE-AAC encoder is operated with a fixed content loudness setting as well, but an audio mixer manually ensures that the long-term integrated content loudness is aligned to the content loudness setting of the xHE-AAC encoder. This workflow might be preferred if the creative intent as to content dynamics and loudness characteristics should be preserved in the best possible way. Note that the "Automatic Leveling Workflow" will always affect the original content dynamics to some degree, based on the individual behavior of the automatic leveling process.

For pre-produced files combined with live sources, it is recommended to pre-normalize the pre-produced content to the same content loudness as used for the leveling workflow.

MPEG-D DRC Decoder Settings

When decoding xHE-AAC bitstreams, the embedded MPEG-D DRC metadata can be applied during playback in a flexible way to enhance the user experience in various playback scenarios. There are two MPEG-D DRC related decoder settings; these can be controlled by an application or by the user, depending on the specific xHE-AAC use case:

The "Target Loudness" setting controls the internal decoder gain, which is used for normalization of all decoded audio content to the same output loudness. The loudness normalization gain is derived based on the encoded content loudness in the xHE-AAC bitstream and the target loudness set at the decoder.

In media streaming applications, the target loudness is usually set to a value of -24 LKFS. When decoding for a multi-channel playback system such as an AVR, the target loudness should be set to a lower level such as -31 LKFS for alignment to other audio decoder sources. For playback on mobile devices, it is recommended to limit the target loudness setting to a maximum of -16 LKFS if a larger output loudness than -24 LKFS is desired.

The "DRC Effect Type" setting controls the application of dynamic range control instructions to the decoded audio content. There are four DRC effect type settings that are relevant for most applications:

- "Noisy Environment": DRC processing for conditions adversely affected by environmental noise.
- "Late Night": DRC processing for scenarios in which e.g. neighbors or kids should not be disturbed by loud parts above the average.
- "Limited Playback Range": DRC processing for listening on mobile devices at high target loudness and limited transducer capabilities.
- "Off": No DRC processing except for peak limiting, which might be required depending on the specific target loudness setting.

It is important to note that the DRC effect type setting is a preference setting for achieving long-term effects on content dynamics. Depending on the specific type of audio content, the algorithms used for generation of dynamic range control instructions at the xHE-AAC encoder, and most importantly the current short-term dynamics, the perceived instantaneous DRC effect might be different.

Except for the "Late Night" setting, it is recommended to control all MPEG-D DRC related decoder settings in an automatic way. The "Noisy Environment" setting could, for example, be enabled based on environmental parameters such as noise level or movement parameters. Optionally, the user could be presented with a slider for increasing the target loudness from -24 LKFS to -16 LKFS in adversely affected environments. In all other cases, internal system control of target loudness and DRC effect type setting is strongly recommended.

USE CASES

File Encoding

The most basic of all audio encoding use cases is the simple encoding of a WAV file, e.g. your favorite song, to an xHE-AAC encoded MP4 file or downloading your daily podcasts to your mobile phone.

Figure 3: File Encoding

The source material is a simple WAV file (PCM audio), which is encoded with xHE-AAC and stored in an MP4 file.

Streaming

The streaming use cases for xHE-AAC cover all means of over-the-top (OTT) delivery, from a live shoutCAST internet radio station to an on-demand video streamed to a mobile phone.

There are two major use case categories – streaming, live and on-demand – which differ mainly in the source of the PCM input and the handling of content loudness.

On-Demand Streaming

Since most streaming services these days use either HLS or MPEG-DASH, an on-demand service is not much different from the above file encoding use case.

Figure 4: On-Demand Streaming

The main difference is the fragmentation of the MP4 file after encoding to enable streaming and the creation of a manifest that describes these fragments.

Live Streaming

For live services, the uncompressed PCM comes from a real-time source, e.g. a sound card. It is generally not possible to measure the content loudness in a live setup and the content is expected to be normalized by external means, e.g. a sound engineer or a real-time loudness leveling device.

Digital Radio

xHE-AAC has been defined as the mandatory audio codec for Digital Radio Mondiale (DRM). In the current version of this document, this particular use case is not further described. Implementation guidance for this specific technical area can be found in the application bulletin "xHE-AAC in Digital Radio Mondiale" [DRMAP].

TECHNOLOGY DETAILS

xHE-AAC Stream Access Point (SAP) and Immediate Playout Frame (IPF)

The approach for supporting bitstream switching in xHE-AAC is very different from older AAC profiles because, unlike for them, the adaptive streaming use case was part of the original requirements list when designing the profile.

Instead of having to impose difficult-to-achieve constraints on the encoding process, xHE-AAC introduces Immediate Playout Frames (IPFs) as an explicit implementation of Stream Access Points (SAPs). IPFs act in a very similar way to I-frames or IDR-frames known from video coding, which is why IPFs are sometimes also referred to as "Audio I-Frames".

Having an explicit implementation of SAPs makes it easier to identify them in the bitstream for verification and compliance in MP4 file fragmentation/segmentation, but it also helps in assuring optimal audio quality during the switching process. Most importantly, however, IPFs allow switching of coding tools while maintaining seamless transitions. This makes it possible to cover the complete bit rate range from 12-500 kbit/s or above in a single Adaptation Set without compromising on audio quality, whereas all other profiles only cover a certain range of bit rates. This greatly simplifies the design of DASH and other adaptive bit rate services.

IPFs allow the creation of audio files and streams that are free of the priming and fade-in effect, which has led to complications and problems of time-alignment and synchronization across media types. In short, "priming-free" means that upon start-up of a decoder instance, the first samples that a decoder produces from the IPF access unit are the reconstructed samples of the first samples to have entered the encoder (as opposed to first outputting a sequence of zero-valued "priming" samples or a fade-in version of the input to the encoder). In a file-to-file encode-decode, the codec thus appears to be "delay-free."

Construction of an IPF

The basic principle for constructing an IPF is illustrated in figure 7. In order to allow immediate playout of audio samples after reception of the current AU(n), it is necessary to include the previous AU(n-1) as an "Audio Pre-Roll" within a so-called extension payload. This is necessary because in a regular decoding process it is assumed that the xHE-AAC decoder is in a certain state when decoding the current AU. This state information can be reconstructed when decoding the Audio Pre-Roll but would be missing otherwise.

In addition, it has to be assured that the bit stream of the current AU and its Audio Pre-Roll can be decoded independently, i.e. without prediction or other dependencies to previous AUs. This is achieved by setting the usacIndependencyFlag ("indepFlag" in figure 7), which resets the bitstream parsing process.

Since the codec configuration structure can be included as part of the extension payload, it is also possible to change the configuration during a switch. Hence, the optimal configuration for each bit rate can be selected without negative (i.e. noticeable) impact on the stream transition. The construction and decoding of an IPF are normatively specified in the MPEG standard; for further reading, please refer to the standard document [USAC]. It is emphasized once more that a standardized and explicit mechanism for bitstream switching is defined for xHE-AAC, which makes it the ideal choice for adaptive streaming services like DASH and HLS.

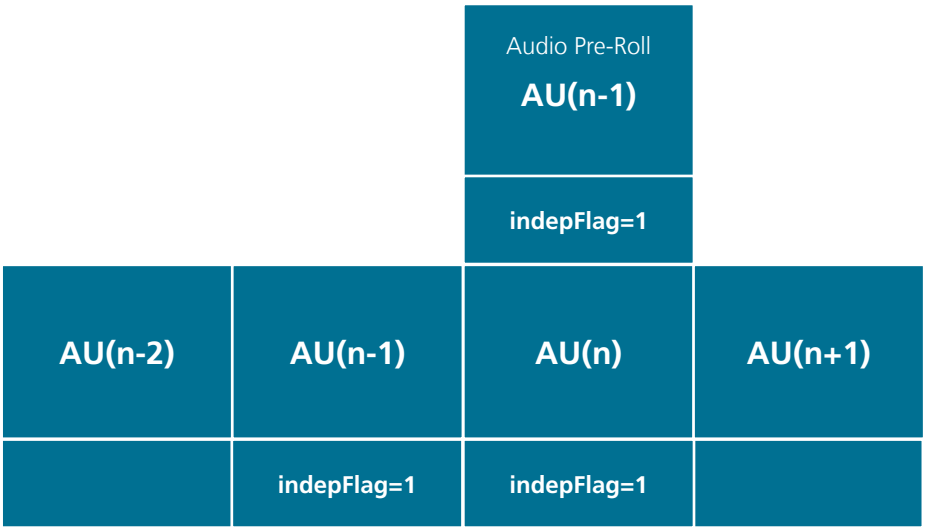

Figure 5 Construction of an Immediate Playout Frame (IPF)

Stream ID

The stream identifier (StreamID) is a simple 16-bit number that is part of each codec configuration structure. It must be supplied for each Representation and it must be different for each Representation within one Adaptation Set. It is a straightforward mechanism to distinguish between configurations that are otherwise identical.

Framing

xHE-AAC can operate in various operation modes, which differ in audio frame size. This frame size is roughly correlated with the bit rate and will vary between Representations (lower bit rates cause larger frame sizes, i.e. durations). In order to keep the segment borders time-aligned across all Representations, the length (in audio samples) of any segment must be a multiple of the least common multiple (LCM) of all frame sizes in that Adaptation Set.

For example, in a typical Adaptation Set there may be a high-rate Representation with 1024 audio sample framing as well as 2048 and 4096 framing for mid- and low-rate Representations. In this case, all segment lengths must be a multiple of 4096, because LCM(1024, 2048, 4096) = 4096. In order to keep things simple, and if possible, it is recommended to always insert IPFs at sample counts that are a multiple of 4096.

Technically, xHE-AAC can also operate on granule lengths of 768 audio samples. However, that mode is not recommended, not least because it increases the abovementioned segment length granularity up to 12,288 audio samples.

CONTAINER FORMATS

LATM/LOAS

When xHE-AAC is stored or transmitted "raw", i.e. without an additional container, multiplexing and streaming is enabled by the MPEG-4 Low Overhead Audio Transport Multiplex (LATM) and the Low Overhead Audio Stream (LOAS) [MP4A]. Support for the alternative Audio Data Transport Stream (ADTS), which was common for AAC-LC and HE-AAC, is deprecated with xHE-AAC and not supported anymore. LATM/LOAS is typically used in SHOUTcast deployments.

File-format (ISOBMFF/MP4FF)

Storage of xHE-AAC in the ISO base media file format (ISOBMFF)[ISOBMFF] follows the same principles as AAC-LC and HE-AAC, i.e. the MP4 file format [MP4FF] is used.
All IPFs are signaled by means of the "SyncSampleBox". IPFs allow the decoder to fully reconstruct the signal without any previous AUs, which enables true random access at any sync sample. This is particularly useful when a flat MP4 file is used as input to a streaming system for subsequent fragmentation. Signaling of IPF is mandatory for xHE-AAC.
Since the xHE-AAC encoder works on a fixed "granule" of e.g. 2048 audio samples, the last AU of an MP4 file usually represents only the last few samples of the original WAV file. In order to restore this original file length, an edit-list can be used to trim the MP4 file accordingly. It is recommended that an xHE-AAC file starts with an IPF, which addresses the "priming" issue (see above) and removes the need for edit lists at the start of the item.

In addition to the rather expensive IPFs, all AUs that have the usacIndependencyFlag set to 1 can be used to enable random access, e.g. for seeking operations. While these Independency Frames (IF) can be used to start decoding, a full audio signal is guaranteed only after decoding a certain number of AUs. This is referred to as roll distance in file format terms and can be signaled using the AudioPreRollEntry and the AudioSampleGroupEntry respectively.

Common Media Application Format – CMAF

CMAF [CMAF] specifies media profiles that are based on general ISOBMFF constraints combined with specific codecs. These media profiles can be combined into presentation profiles that essentially provide the "wire format" for DASH and HLS services. The xHE-AAC media profile is identified by the cxha brand.

MPEG-2 TS

When xHE-AAC is encapsulated in an MPEG-2 Transport Stream (MPEG-2 TS) [MP2TS], LATM/LOAS frames are encapsulated in PES packets. Since these frames carry in-band configuration information by means of the StreamMuxConfig element, no additional configuration information is required. However, the xHE-AAC level may be signaled in the MPEG-4_audio_descriptor (0x68 – 0x6D).

MIME/Media Types
It is common for HTTP based services (streaming and download) to use the MIME Type parameter to signal additional properties of the content, e.g. in the Content-Type field of the HTTP header and/or as part of a manifest file. The following Media Types are relevant for use with xHE-AAC:

- audio/usac for LATM/LOAS
- audio/mp4 for audio-only MP4 files
- video/mp4 for multiplexed MP4 files

It is possible to signal additional content properties defined in RFC 6831 for "Bucket Media Types". This mechanism makes it possible to expose the codecs that are inside the container. For xHE-AAC in an MP4 file, the complete Content-Type is:

- Content-Type: audio/mp4; codecs=mp4a.40.42

The codecs parameter is used in both DASH MPDs and HLS Playlists. Media Types are properties of the content and are usually part of the HTTP protocol but may be signaled in DASH MPDs.

OTT STREAMING

DASH

DASH streaming of xHE-AAC is based on CMAF file fragments together with a manifest, the Media Presentation Description (MPD). The file is essentially split into multiple segments, which can be individually addressed by a URL, potentially using additional information such as segment indexing and byte-range requests. The details of DASH are specified in MPEG-DASH [DASH] and the DASH Industry Forum's Interoperability Guidelines [DASHIF].

Figure 6: DASH streaming

In the most general case, the output from the xHE-AAC encoder is still a flat MP4 file, which will then be processed by a segmenter. The segmenter uses the information from the ISOMBMFF SyncSampleBox (IPF locations) together with additional configuration, like preferred segment duration, to create the DASH segments. Some implementations might choose to integrate the segmenter as part of the encoder. For live services, the input to the segmenter might be just a (self-contained) fragment of an MP4 file.

For xHE-AAC content, DASH-IF specifies the following Interoperability Point (IOP):

http://dashif.org/guidelines/dashif#usac

supporting xHE-AAC up to 5.1 multichannel. All DASH segments start with an IPF to enable bit rate switching and random access. Additional signaling may be required by means of the MIME Type (see above).

HLS

HTTP Live Streaming (HLS) is specified in RFC 8261 [HLS] and can be used in combination with CMAF for both live and on-demand streaming services. The content segmentation is the same as described above for DASH.

The main difference between DASH and HLS is the format of the manifest, which is called a Playlist for HLS. The CODECS attribute is used to specify the codecs that are used in the presentation. The attribute uses the RFC 6381 Content-Type format (see above); for xHE-AAC, this is:

- CODECS="mp4a.40.42"

GLOSSARY

AAC	Advanced Audio Coding, a family of high-quality audio coding schemes
AAC-LC	AAC Low Complexity AOT 2, sometimes used synonymously with the AAC profile
ADTS	Audio Data Transport Stream, a framing mechanism for legacy AAC
AOT	Audio Object Type, identifying a specific MPEG-4 Audio coding scheme or tool
AU	Access Unit, smallest part of the encoded bitstream that can be decoded by itself
DASH	Dynamic Adaptive Streaming over HTTP, an adaptive streaming format
DRC	Dynamic Range Control
HE-AAC	High Efficiency AAC profile, AOT 2 (AAC LC) and AOT 5 (SBR)
HE-AACv2	High Efficiency AACv2 profile, AOT 2, 5 and 29 (PS)
HLS	HTTP Live Streaming, an adaptive streaming format
HTTP	Hypertext Transfer Protocol, a widely deployed Internet protocol
IPF	Immediate Playout Frame, an independent and switchable audio frame
ISOBMFF	ISO base media file format, the base file format for the MP4 file format
LATM	Low-overhead MPEG-4 Audio Transport Multiplex
LOAS	Low Overhead MPEG-4 Audio Stream
m3u8	HLS playlist format
MP3	MPEG-1/2 Layer 3 audio codec
MP4 file	A file formatted according to the MP4 file format
MP4 file format	A file format based on ISOBMFF tailored for use with MPEG-4 profiles and codecs
MPD	Media Presentation Description, the DASH manifest format
MPEG	Moving Picture Experts Group, the informal name of ISO/IEC JTC1 1/SC 29/WG 11
PCM	Pulse-code modulation, the standard representation of uncompressed digital audio
PS	Parametric Stereo, an MPEG-4 Audio coding tool
SBR	Spectral Band Replication, an MPEG-4 Audio coding tool
USAC	Unified Speech and Audio Coding, an MPEG-D coding tool
WAV file	Audio file format to store PCM
xHE-AAC	Extended High Efficiency AAC, an AAC profile

REFERENCES

MPEG2TS	ISO/IEC 13818-1 Generic coding of moving pictures and associated audio information: Systems
MP4A	ISO/IEC 14496-3 Coding of audio-visual objects – Part 3: Audio
ISOBMFF	ISO/IEC 14496-3 Coding of audio-visual objects – Part 12: ISO base media file format
MP4FF	ISO/IEC 14496-3 Coding of audio-visual objects – Part 14: MP4 file format
USAC	ISO/IEC 23003-3 MPEG audio technologies – Part 3: Unified speech and audio coding
DRC	ISO/IEC 23003-4 MPEG audio technologies – Part 4: Dynamic range control
DASH	ISO/IEC 23009-1 Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats
DASHIF	Guidelines for Implementation: DASH-IF Interoperability Points
HLS	RFC 8216 HTTP Live Streaming
CMAF	ISO/IEC 23000-19 Coding of audio-visual objects – Part 19: Common media application format (CMAF) for segmented media
DRMAP	https://www.iis.fraunhofer.de/content/dam/iis/de/doc/ame/wp/FraunhoferIIS_ApplicationBulletin_xHE-AACinDRM.pdf

ABOUT FRAUNHOFER IIS

The Audio and Media Technologies division of Fraunhofer IIS has been an authority in its field for more than 25 years, starting with the creation of mp3 and co-development of AAC formats. Today, almost all consumer electronic devices, computers and mobile phones are equipped with Fraunhofer's media technologies. Besides the global successes mp3 and AAC, the Fraunhofer technologies that improve consumers' audio experiences include Cingo® (spatial VR audio), Symphoria® (automotive 3D audio), xHE-AAC (adaptive streaming and digital radio), the 3GPP EVS VoLTE codec (crystal clear telephone calls), and the interactive and immersive MPEG-H TV Audio System.

With the test plan for the Digital Cinema Initiative and the recognized software suite easyDCP, Fraunhofer IIS significantly pushed the digitization of cinema. The most recent technological achievement for moving pictures is Realception®, a tool for light-field data processing.

Fraunhofer IIS, based in Erlangen, Germany, is one of 72 institutes and research units of Fraunhofer-Gesellschaft, Europe's largest application-oriented research organization.

For more information, contact amm-info@iis.fraunhofer.de or visit www.iis.fraunhofer.de/amm.