Extended Audio File Services Reference

Audio > Core Audio

2008-09-09

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

Extended Audio File Services Reference

Framework:	AudioToolbox/AudioToolbox.h
Declared in	ExtendedAudioFile.h

Overview

Extended Audio File Services provides simplified audio file access, combining features of Audio File Services and Audio Converter Services. It provides a unified interface for reading and writing compressed as well as linear PCM audio files.

Functions by Task

Managing Extended Audio File Objects

`ExtAudioFileCreateWithURL` (page 7)
Creates a new audio file and associates it with a new extended audio file object.

`ExtAudioFileDispose` (page 8)
Disposes of an extended audio file object and closes the associated file.

`ExtAudioFileOpenURL` (page 11)
Opens an existing audio file for reading, and associates it with a new extended audio file object.

`ExtAudioFileWrapAudioFileID` (page 14)
Wraps an audio file object in an extended audio file object.

Configuring Properties for Extended Audio File Objects

`ExtAudioFileGetProperty` (page 9)
Gets a property value from an extended audio file object.

`ExtAudioFileGetPropertyInfo` (page 9)
Gets information about an extended audio file object property.

`ExtAudioFileSetProperty` (page 12)
Sets a property value for an extended audio file object.

Reading and Writing Audio Data

`ExtAudioFileRead` (page 11)

Performs a synchronous, sequential read operation on an audio file.

`ExtAudioFileSeek` (page 12)

Seeks to a specified frame in a file.

`ExtAudioFileTell` (page 13)

Gets an audio file's read/write position.

`ExtAudioFileWrite` (page 14)

Performs a synchronous, sequential write operation on an audio file.

`ExtAudioFileWriteAsync` (page 15)

Perform an asynchronous, sequential write operation on an audio file.

Deprecated Functions

`ExtAudioFileCreateNew` (page 6)

Deprecated. Use the `ExtAudioFileCreateWithURL` (page 7) function instead.

`ExtAudioFileOpen` (page 10)

Deprecated. Use the `ExtAudioFileOpenURL` (page 11) function instead.

Functions

ExtAudioFileCreateNew

Deprecated. Use the `ExtAudioFileCreateWithURL` (page 7) function instead.

```
 ExtAudioFileCreateNew(
    const struct FSRef                  *inParentDir,
    CFStringRef                         inFileName,
    AudioFileTypeID                     inFileType,
    const AudioStreamBasicDescription   *inStreamDesc,
    const AudioChannelLayout            *inChannelLayout,
    ExtAudioFileRef                     *outExtAudioFile
);
```

Parameters

`inParentDir`

The directory in which to create the new file.

`inFileName`

The name of the new file.

`inFileType`

The type of file to create. This is a constant from AudioToolbox/AudioFile.h, e.g. kAudioFileAIFFType. Note that this is not an HFSTypeCode.

`inStreamDesc`

The format of the audio data to be written to the file.

inChannelLayout

The channel layout of the audio data. If non-null, this must be consistent with the number of channels specified by inStreamDesc.

outExtAudioFile

On output, a newly allocated extended audio file object.

Return Value

A result code.

Discussion

Creates a new audio file.

If the file to be created is in a compressed format, it is permissible for the sample rate in inStreamDesc to be 0, since in all cases, the file's encoding AudioConverter may produce audio at a different sample rate than the source. The file will be created with the audio format actually produced by the encoder.

Special Considerations

This function is deprecated. Use the `ExtAudioFileCreateWithURL` (page 7) function instead.

Availability

Available in Mac OS X v10.4 and later.

Related Sample Code

RecordAudioToFile

Declared In

`ExtendedAudioFile.h`

ExtAudioFileCreateWithURL

Creates a new audio file and associates it with a new extended audio file object.

```
OSStatus ExtAudioFileCreateWithURL (
    CFURLRef                          inURL,
    AudioFileTypeID                   inFileType,
    const AudioStreamBasicDescription *inStreamDesc,
    const AudioChannelLayout          *inChannelLayout,
    UInt32                            inFlags,
    ExtAudioFileRef                   *outExtAudioFile
);
```

Parameters

inURL

The URL of the new audio file.

inFileType

The type of file to create, specified as a constant from the `Built-In Audio File Types` enumeration.

inStreamDesc

The format of the audio data to be written to the file.

inChannelLayout

The channel layout of the audio data. If non-null, this must be consistent with the number of channels specified by the *inStreamDesc* parameter.

inFlags

Flags for creating or opening the file. If the `kAudioFileFlags_EraseFile` flag is set, it erases an existing file. If the flag is not set, the function fails fails if the URL points to an existing file.

outExtAudioFile

On output, a newly allocated extended audio file object.

Return Value

A result code.

Discussion

If the file to be created is in a compressed format, you may set the sample rate in the *inStreamDesc* parameter to `0`. In all cases, the extended file object's encoding converter may produce audio at a different sample rate than the source. The file will be created with the audio format produced by the encoder.

Availability

Available in Mac OS X v10.5 and later.

See Also

`ExtAudioFileDispose` (page 8)

`ExtAudioFileOpenURL` (page 11)

Declared In

`ExtendedAudioFile.h`

ExtAudioFileDispose

Disposes of an extended audio file object and closes the associated file.

```
OSStatus ExtAudioFileDispose (
    ExtAudioFileRef inExtAudioFile
);
```

Parameters

inExtAudioFile

The extended audio file object to close.

Return Value

A result code.

Availability

Available in Mac OS X v10.4 and later.

See Also

`ExtAudioFileCreateWithURL` (page 7)

Related Sample Code

RecordAudioToFile

Declared In

`ExtendedAudioFile.h`

ExtAudioFileGetProperty

Gets a property value from an extended audio file object.

```
OSStatus ExtAudioFileGetProperty (
    ExtAudioFileRef         inExtAudioFile,
    ExtAudioFilePropertyID  inPropertyID,
    UInt32                  *ioPropertyDataSize,
    void                    *outPropertyData
);
```

Parameters

inExtAudioFile

The extended audio file object to get a property value from.

inPropertyID

The property whose value you want.

ioPropertyDataSize

On input, the size of the memory pointed to by the *outPropertyData* parameter. On output, the size of the property value.

outPropertyData

On output, the property value you wanted to get.

Return Value

A result code.

Availability

Available in Mac OS X v10.4 and later.

See Also

`ExtAudioFileGetPropertyInfo` (page 9)

`ExtAudioFileSetProperty` (page 12)

Related Sample Code

RecordAudioToFile

Declared In

`ExtendedAudioFile.h`

ExtAudioFileGetPropertyInfo

Gets information about an extended audio file object property.

```
OSStatus ExtAudioFileGetPropertyInfo (
    ExtAudioFileRef         inExtAudioFile,
    ExtAudioFilePropertyID  inPropertyID,
    UInt32                  *outSize,
    Boolean                 *outWritable
);
```

Parameters

inExtAudioFile

The extended audio file object to get property information from.

inPropertyID

The property you want information about.

outSize

On output, the size of the property value in bytes. Can be `NULL` on output.

outWritable

On output, a Boolean value indicating whether the property value is writable (`true` means writable). Can be `NULL` on output.

Return Value

A result code.

Availability

Available in Mac OS X v10.4 and later.

See Also

`ExtAudioFileGetProperty` (page 9)

`ExtAudioFileSetProperty` (page 12)

Declared In

`ExtendedAudioFile.h`

ExtAudioFileOpen

Deprecated. Use the `ExtAudioFileOpenURL` (page 11) function instead.

```
 ExtAudioFileOpen (
    const struct FSRef  *inFSRef,
    ExtAudioFileRef     *outExtAudioFile
);
```

Parameters

inFSRef

The audio file to read.

outExtAudioFile

On exit, a newly allocated ExtAudioAudioFileRef.

Return Value

A result code.

Discussion

Allocates a new ExtAudioFileRef for reading an existing audio file.

Special Considerations

This function is deprecated. Use the `ExtAudioFileOpenURL` (page 11) function instead.

Availability

Available in Mac OS X v10.4 and later.

Declared In

`ExtendedAudioFile.h`

ExtAudioFileOpenURL

Opens an existing audio file for reading, and associates it with a new extended audio file object.

```
OSStatus ExtAudioFileOpenURL (
    CFURLRef         inURL,
    ExtAudioFileRef  *outExtAudioFile
);
```

Parameters

inURLRef

The audio file to read.

outExtAudioFile

On output, a newly allocated extended audio file object.

Return Value

A result code.

Availability

Available in Mac OS X v10.5 and later.

See Also

ExtAudioFileCreateWithURL (page 7)

ExtAudioFileDispose (page 8)

Declared In

ExtendedAudioFile.h

ExtAudioFileRead

Performs a synchronous, sequential read operation on an audio file.

```
OSStatus ExtAudioFileRead (
    ExtAudioFileRef  inExtAudioFile,
    UInt32           *ioNumberFrames,
    AudioBufferList  *ioData
);
```

Parameters

inExtAudioFile

The extended audio file object that represents the file you want to read.

ioNumberFrames

On input, the number of frames to read from the file. On output, the number of frames actually read. Fewer frames may be read than were requested. For example, the supplied buffers may not be large enough to accommodate the requested data. If 0 frames are returned, end-of-file was reached.

ioData

One or more buffers into which the audio data is read.

Return Value

A result code.

Discussion

If the extended audio file object has an application data format, then the object's converter converts the file data to the application format.

This function works only on a single thread. If you want your application to read an audio file on multiple threads, use Audio File Services instead.

Availability
Available in Mac OS X v10.4 and later.

See Also
`ExtAudioFileWrite` (page 14)
`ExtAudioFileWriteAsync` (page 15)

Declared In
`ExtendedAudioFile.h`

ExtAudioFileSeek

Seeks to a specified frame in a file.

```
OSStatus ExtAudioFileSeek (
    ExtAudioFileRef  inExtAudioFile,
    SInt64           inFrameOffset
);
```

Parameters

inExtAudioFile
The extended audio file object that represents the file you are working with.

inFrameOffset
The desired seek position, in sample frames, relative to the beginning of the file. Seek position is specified in the sample rate and frame count of the file's audio data format—not your application's audio data format.

Return Value
A result code.

Discussion
Sets the file's read position to the specified sample frame number. A subsequent call to the `ExtAudioFileRead` (page 11) function returns samples from precisely this location, even if it is located in the middle of a packet.

Ensure that the file you are seeking in is open for reading only. This function's behavior with files open for writing is undefined.

Availability
Available in Mac OS X v10.4 and later.

See Also
`ExtAudioFileTell` (page 13)

Declared In
`ExtendedAudioFile.h`

ExtAudioFileSetProperty

Sets a property value for an extended audio file object.

```
OSStatus ExtAudioFileSetProperty (
    ExtAudioFileRef         inExtAudioFile,
    ExtAudioFilePropertyID  inPropertyID,
    UInt32                  inPropertyDataSize,
    const void              *inPropertyData
);
```

Parameters

inExtAudioFile

The extended audio file object to set a property value on.

inPropertyID

The property whose value you want to set.

inPropertyDataSize

The size of the property value, in bytes.

inPropertyData

The value you want to apply to the specified property.

Return Value

A result code.

Availability

Available in Mac OS X v10.4 and later.

See Also

`ExtAudioFileGetProperty` (page 9)

`ExtAudioFileGetPropertyInfo` (page 9)

Related Sample Code

RecordAudioToFile

Declared In

`ExtendedAudioFile.h`

ExtAudioFileTell

Gets an audio file's read/write position.

```
OSStatus ExtAudioFileTell (
    ExtAudioFileRef  inExtAudioFile,
    SInt64           *outFrameOffset
);
```

Parameters

inExtAudioFile

The extended audio file object that represents the file you are working with.

outFrameOffset

On output, the file's current read/write position in sample frames. Read/write position is specified in the sample rate and frame count of the file's audio data format—not your application's audio data format.

Return Value

A result code.

Availability
Available in Mac OS X v10.4 and later.

See Also
`ExtAudioFileSeek` (page 12)

Declared In
`ExtendedAudioFile.h`

ExtAudioFileWrapAudioFileID

Wraps an audio file object in an extended audio file object.

```
OSStatus ExtAudioFileWrapAudioFileID (
    AudioFileID      inFileID,
    Boolean          inForWriting,
    ExtAudioFileRef  *outExtAudioFile
);
```

Parameters

`inFileID`

The audio file object to wrap.

`inForWriting`

Use `true` if you intend to write to the audio file, `false` otherwise.

`outExtAudioFile`

On output, a newly allocated extended audio file object.

Return Value
A result code.

Discussion
Allocates a new extended audio file object that wraps an existing audio file object. Your application is responsible for keeping the audio file object open until the extended audio file object is disposed.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`ExtendedAudioFile.h`

ExtAudioFileWrite

Performs a synchronous, sequential write operation on an audio file.

```
OSStatus ExtAudioFileWrite (
    ExtAudioFileRef        inExtAudioFile,
    UInt32                 inNumberFrames,
    const AudioBufferList  *ioData
);
```

Parameters

`inExtAudioFile`

The extended audio file object that represents the file to write to.

inNumberFrames

The number of frames to write.

ioData

The buffer(s) from which audio data is written to the file.

Return Value

A result code.

Discussion

If the extended audio file object has an application data format, then the object's converter converts the data in the *ioData* parameter to the file data format.

Availability

Available in Mac OS X v10.4 and later.

See Also

`ExtAudioFileRead` (page 11)

`ExtAudioFileWriteAsync` (page 15)

Declared In

`ExtendedAudioFile.h`

ExtAudioFileWriteAsync

Perform an asynchronous, sequential write operation on an audio file.

```
OSStatus ExtAudioFileWriteAsync (
    ExtAudioFileRef        inExtAudioFile,
    UInt32                 inNumberFrames,
    const AudioBufferList  *ioData
);
```

Parameters

inExtAudioFile

The extended audio file object that represents the file you want to write to.

inNumberFrames

The number of frames to write.

ioData

The buffer(s) from which audio data is written to the file.

Return Value

A result code.

Discussion

Writes the provided buffer list to an internal ring buffer and notifies an internal thread to perform the write at a later time. The first time this function is called, allocations may be performed. You can call this function with `0` frames and a `NULL` buffer in a non-time-critical context to initialize the asynchronous mechanism. Once initialized, subsequent calls are very efficient and do not take locks. This technique may be used to write to a file from a realtime thread.

Your application must not mix synchronous and asynchronous writes to the same file.

Pending writes are not guaranteed to be flushed to disk until the `ExtAudioFileDispose` (page 8) function is called.

Errors may occur after this call has returned. Such errors may be returned from subsequent calls to this function.

Availability
Available in Mac OS X v10.4 and later.

See Also
`ExtAudioFileRead` (page 11)
`ExtAudioFileWrite` (page 14)

Related Sample Code
RecordAudioToFile

Declared In
`ExtendedAudioFile.h`

Data Types

ExtAudioFileRef

An opaque structure representing an extended audio file object.

```
typedef struct OpaqueExtAudioFile *ExtAudioFileRef;
```

Availability
Available in Mac OS X v10.4 and later.

Declared In
`ExtendedAudioFile.h`

ExtAudioFilePropertyID

An audio file object property identifier.

```
typedef UInt32 ExtAudioFilePropertyID;
```

Availability
Available in Mac OS X v10.4 and later.

Declared In
`ExtendedAudioFile.h`

Constants

Property Identifiers for Extended Audio File Objects

```
enum {
    kExtAudioFileProperty_FileDataFormat       = 'ffmt',
    kExtAudioFileProperty_FileChannelLayout    = 'fclo',
    kExtAudioFileProperty_ClientDataFormat     = 'cfmt',
    kExtAudioFileProperty_ClientChannelLayout  = 'cclo'

    // read-only properties:
    kExtAudioFileProperty_AudioConverter       = 'acnv',
    kExtAudioFileProperty_AudioFile            = 'afil',
    kExtAudioFileProperty_FileMaxPacketSize    = 'fmps',
    kExtAudioFileProperty_ClientMaxPacketSize  = 'cmps',
    kExtAudioFileProperty_FileLengthFrames     = '#frm',

    // read/write properties:
    kExtAudioFileProperty_ConverterConfig      = 'accf',
    kExtAudioFileProperty_IOBufferSizeBytes    = 'iobs',
    kExtAudioFileProperty_IOBuffer             = 'iobf'
};
```

Constants

`kExtAudioFileProperty_FileDataFormat`

A read-only `AudioStreamBasicDescription` structure. Represents a file's data format.

Available in Mac OS X v10.4 and later.

Declared in `ExtendedAudioFile.h`.

`kExtAudioFileProperty_FileChannelLayout`

A read/write `AudioChannelLayout` structure.

When writing, the channel layout is written to the file, if the format specified in the `kExtAudioFileProperty_FileDataFormat` property supports the layout. If the format does not support the layout, the channel layout is still interpreted as the destination layout when performing conversion from the client channel layout, if any.

When reading, the specified layout overrides the one read from the file, if one is present in the file.

You must set this property before setting the application audio data format or application channel layout in the extended audio file object.

Available in Mac OS X v10.4 and later.

Declared in `ExtendedAudioFile.h`.

`kExtAudioFileProperty_ClientDataFormat`

A read/write `AudioStreamBasicDescription` structure.

The format must be linear PCM (specified by the `kAudioFormatLinearPCM` constant from the `CoreAudioTypes.h` header file). You must set this property to allow encoding or decoding of a non-PCM file data format. You can set this property on PCM files to specify the data format to use in your read and write calls.

Available in Mac OS X v10.4 and later.

Declared in `ExtendedAudioFile.h`.

kExtAudioFileProperty_ClientChannelLayout

A read/write AudioChannelLayout structure.

Specifies the channel layout of the buffers in an AudioBufferList list that you pass to the ExtAudioFileRead (page 11) and ExtAudioFileWrite (page 14) functions. This layout may be different from the file's channel layout, in which case the extended audio file object's underlying audio converter performs remapping. This property must be set after setting the kExtAudioFileProperty_ClientDataFormat property, and the number of channels in the two layouts must match.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_AudioConverter

A read-only property whose value is the audio converter object that is associated with the extended audio file object.

If you alter any properties of the audio converter—the bit rate, for instance—you must then set the kExtAudioFileProperty_ConverterConfig property. When you do so, using a NULL configuration is sufficient. Setting that property ensure that the output file's data format is consistent with the format being produced by the converter.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_AudioFile

A read-only property whose value is the audio file object that is associated with the extended audio file object.

Available in Mac OS X v10.5 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_FileMaxPacketSize

A read-only UInt32 value that represents the file data format's maximum packet size, in bytes.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_ClientMaxPacketSize

A read-only UInt32 value that represents your application's data format's maximum packet size, in bytes.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_FileLengthFrames

An SInt64 value that represents the associated audio file's length in sample frames. Read-only for non-PCM format files; read/write for files in PCM formats.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

kExtAudioFileProperty_ConverterConfig

A read/write CFArray object that represents the underlying audio converter's configuration, as specified by the kAudioConverterPropertySettings property.

Set this property's value to NULL to force resynchronization of the converter's output format with the file's data format.

Available in Mac OS X v10.4 and later.

Declared in ExtendedAudioFile.h.

`kExtAudioFileProperty_IOBufferSizeBytes`

A read/write `UInt32` value that represents the size of the buffer that the underlying converter object uses to read or write the associated audio file. This property has a value when there is an underlying audio converter object present.

Available in Mac OS X v10.4 and later.

Declared in `ExtendedAudioFile.h`.

`kExtAudioFileProperty_IOBuffer`

A read/write `void*` value that refers to an audio data buffer.

This is the memory that the extended audio file object uses for disk I/O when converting between the application and file data formats. You may want your application to share this memory among multiple extended audio file objects. If so, you can set this property to point to a buffer you specify—pass a pointer to a pointer when calling the `ExtAudioFileSetProperty` function. After setting this property, your application must then set the `kExtAudioFileProperty_IOBufferSizeBytes` property.

Available in Mac OS X v10.4 and later.

Declared in `ExtendedAudioFile.h`.

Result Codes

This table lists result codes defined for Extended Audio File Services.

Result Code	Value	Description
`kExtAudioFileError_InvalidProperty`	-66561	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_InvalidPropertySize`	-66562	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_NonPCMClientFormat`	-66563	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_InvalidChannelMap`	-66564	The number of channels does not match the specified format. Available in Mac OS X v10.4 and later.
`kExtAudioFileError_InvalidOperationOrder`	-66565	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_InvalidDataFormat`	-66566	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_MaxPacketSizeUnknown`	-66567	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_InvalidSeek`	-66568	An attempt to write, or an offset, is out of bounds. Available in Mac OS X v10.4 and later.
`kExtAudioFileError_AsyncWriteTooLarge`	-66569	Available in Mac OS X v10.4 and later.
`kExtAudioFileError_AsyncWriteBufferOverflow`	-66570	An asynchronous write operation could not be completed in time. Available in Mac OS X v10.4 and later.

Document Revision History

This table describes the changes to *Extended Audio File Services Reference*.

Date	Notes
2008-09-09	New document that describes a programming interface for reading and writing linear PCM and compressed audio files.

Index