AVFoundation / AVAsset

Class

AVAsset

An object that models timed audiovisual media.

iOS 4.0+ | iPadOS 4.0+ | Mac Catalyst 13.1+ | macOS 10.7+ | tvOS 9.0+ | visionOS 1.0+ | watchOS 1.0+
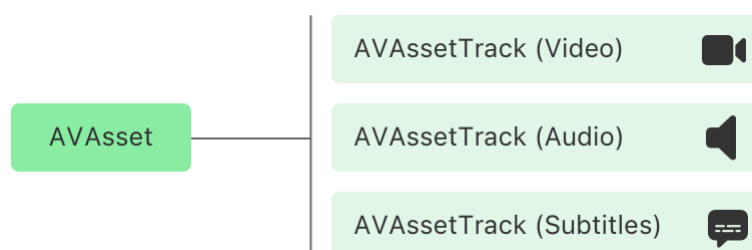
```
class AVAsset
```

Mentioned in

▤ Loading media data asynchronously

▤ Controlling the transport behavior of a player

▤ Exporting video to alternative formats

▤ Implementing simple enhanced buffering for your content

▤ Retrieving media metadata

Overview

An asset models file-based media like a QuickTime movie or an MP3 audio file, and also media streamed using HTTP Live Streaming (HLS). An asset is a container object for one or more instances of AVAssetTrack that model the uniformly typed tracks of media. The most commonly used track types are audio and video, but assets may also contain supplementary tracks, like closed captions, subtitles, and timed metadata.

You load the tracks for an asset by asynchronously loading its <u>tracks</u> property. In some cases, you may want to perform operations on a subset of an asset's tracks rather than on its complete collection. For those situations, an asset provides methods to retrieve subsets of tracks according to particular criteria, such as identifier, media type, or characteristic.

Topics

Creating an asset

~~convenience init(url: URL)~~

Creates an asset that models the media at the specified URL.

Deprecated

Loading duration and timing

static var duration: AVAsyncProperty<Root, CMTime>

A time value that represents the duration of the asset.

static var providesPreciseDurationAndTiming: AVAsyncProperty<Root, Bool>

A Boolean value that indicates whether the asset provides precise duration and timing.

static var minimumTimeOffsetFromLive: AVAsyncProperty<Root, CMTime>

A time value that indicates how closely playback follows the latest live stream content.

Loading tracks

static var tracks: AVAsyncProperty<Root, [AVAssetTrack]>

The tracks of media that an asset contains.

func loadTrack(withTrackID: CMPersistentTrackID, completionHandler: (AVAssetTrack?, (any Error)?) -> Void)

Loads a track that contains the specified identifier.

func loadTracks(withMediaType: AVMediaType, completionHandler: ([AVAssetTrack]?, (any Error)?) -> Void)

Loads tracks that contain media of a specified type.

```
func loadTracks(withMediaCharacteristic: AVMediaCharacteristic,
completionHandler: ([AVAssetTrack]?, (any Error)?) -> Void)
```
Loads tracks that contain media of a specified characteristic.

```
func findUnusedTrackID(completionHandler: (CMPersistentTrackID, (any
Error)?) -> Void)
```
Loads an identifier that no other track in the asset uses.

Loading track groups

```
static var trackGroups: AVAsyncProperty<Root, [AVAssetTrackGroup]>
```
The track groups an asset contains.

Loading metadata

```
static var metadata: AVAsyncProperty<Root, [AVMetadataItem]>
```
The metadata items that an asset contains for all metadata identifiers.

```
static var commonMetadata: AVAsyncProperty<Root, [AVMetadataItem]>
```
The metadata items that an asset contains for common metadata identifiers.

```
static var availableMetadataFormats: AVAsyncProperty<Root, [AVMetadata
Format]>
```
The formats of metadata that an asset contains.

```
func loadMetadata(for: AVMetadataFormat, completionHandler: ([
AVMetadataItem]?, (any Error)?) -> Void)
```
Loads an array of metadata items that the asset contains for the specified format.

```
static var creationDate: AVAsyncProperty<Root, AVMetadataItem?>
```
A metadata item that indicates the creation date of an asset.

```
static var lyrics: AVAsyncProperty<Root, String?>
```
The lyrics of the asset in a language suitable for the current locale.

Loading suitability

Query the suitability properties to determine an asset's support for various purposes, even if only
under a specific set of conditions.

```
static var isPlayable: AVAsyncProperty<Root, Bool>
```

A Boolean value that indicates whether an asset contains playable content.

`static var isExportable: AVAsyncProperty<Root, Bool>`

A Boolean value that indicates whether you can export an asset using an export session.

`static var isReadable: AVAsyncProperty<Root, Bool>`

A Boolean value that indicates whether you can extract the asset's media data using an asset reader.

`static var isComposable: AVAsyncProperty<Root, Bool>`

A Boolean value that indicates whether you can use the asset in a media composition.

`static var isCompatibleWithAirPlayVideo: AVAsyncProperty<Root, Bool>`

A Boolean value that indicates whether the asset is compatible with AirPlay Video.

`static var isCompatibleWithSavedPhotosAlbum: AVAsyncProperty<Root, Bool>`

A Boolean value that indicates whether you can write the asset to the Saved Photos album.

Loading asset preferences

`static var preferredRate: AVAsyncProperty<Root, Float>`

The asset's rate preference for playing its media.

`static var preferredVolume: AVAsyncProperty<Root, Float>`

The asset's volume preference for playing its audible media.

`static var preferredTransform: AVAsyncProperty<Root, CGAffineTransform>`

The asset's transform preference to apply to its visual content during presentation or processing.

`static var preferredDisplayCriteria: AVAsyncProperty<Root, AVDisplayCriteria>`

The asset's display mode preference for optimal playback of its content.

`class AVDisplayCriteria`

An object the system uses to guide the selection of a display mode in tvOS.

Loading media selections

```
static var allMediaSelections: AVAsyncProperty<Root, [AVMediaSelection
]>
```
The available media selections for an asset.

```
static var preferredMediaSelection: AVAsyncProperty<Root, AVMedia
Selection>
```
The default media selections for the media selection groups of an asset.

```
static var availableMediaCharacteristicsWithMediaSelectionOptions:
AVAsyncProperty<Root, [AVMediaCharacteristic]>
```
The media characteristics that provide media selection options.

```
func loadMediaSelectionGroup(for: AVMediaCharacteristic, completion
Handler: (AVMediaSelectionGroup?, (any Error)?) -> Void)
```
Loads a media selection group that contains one or more options with the specified media characteristic.

Loading chapter metadata

```
static var availableChapterLocales: AVAsyncProperty<Root, [Locale]>
```
The locales of an asset's chapter metadata.

```
func loadChapterMetadataGroups(withTitleLocale: Locale, containingItems
WithCommonKeys: [AVMetadataKey]) async throws -> [AVTimedMetadataGroup]
```
Loads chapter metadata that contains the specified title locale and common keys.

```
func loadChapterMetadataGroups(bestMatchingPreferredLanguages: [String
], completionHandler: ([AVTimedMetadataGroup]?, (any Error)?) -> Void)
```
Loads chapter metadata with a locale that best matches the list of preferred languages.

Loading content protections

```
static var hasProtectedContent: AVAsyncProperty<Root, Bool>
```
A Boolean value that indicates whether the asset contains protected content.

Loading fragment support

```
static var canContainFragments: AVAsyncProperty<Root, Bool>
```
A Boolean value that indicates whether you can extend the asset by fragments.

```
static var containsFragments: AVAsyncProperty<Root, Bool>
```
A Boolean value that indicates whether at least one movie fragment extends the asset.

```
static var overallDurationHint: AVAsyncProperty<Root, CMTime>
```
A hint to the total duration of fragments that currently exist or may exist in the future.

Canceling property loading

```
func cancelLoading()
```
Cancels all pending requests to asynchronously load property values.

Retrieving reference restrictions

```
var referenceRestrictions: AVAssetReferenceRestrictions
```
The restrictions that an asset places on how it resolves references to external media.

```
struct AVAssetReferenceRestrictions
```
Restrictions to use when resolving references to external media data.

Deprecated

≔ Deprecated symbols

Review unsupported symbols and their replacements.

Relationships

Inherits From

```
NSObject
```

Inherited By

```
AVComposition, AVMovie, AVURLAsset
```

Conforms To

```
AVAsynchronousKeyValueLoading
```

CVarArg
CustomDebugStringConvertible
CustomStringConvertible
Equatable
Hashable
NSCopying
NSObjectProtocol

See Also

Assets

`class` `AVURLAsset`

An asset that represents media at a local or remote URL.

`class` `AVAssetTrack`

An object that models a track of media that an asset contains.

`class` `AVAssetTrackSegment`

An object that represents a time range segment of an asset track.

`class` `AVAssetTrackGroup`

A group of related tracks in an asset.