

[AVFoundation](#) / [AVError](#)

Structure

AVError

A structure that defines the errors that framework operations can generate.

iOS | iPadOS | Mac Catalyst | macOS | tvOS | visionOS | watchOS

```
struct AVError
```

Topics

Error domain

```
static var errorDomain: String
```

Error codes

```
enum Code
```

An enumeration that defines the errors that framework operations can generate.

```
static var airPlayControllerRequiresInternet: AVError.Code
```

The AirPlay controller requires an internet connection to function.

```
static var airPlayReceiverRequiresInternet: AVError.Code
```

The AirPlay receiver requires an internet connection to function.

```
static var airPlayReceiverTemporarilyUnavailable: AVError.Code
```

An AirPlay receiver is temporarily unavailable.

```
static var applicationIsNotAuthorizedToUseDevice: AVError.Code
```

The user denied this app permission to capture media.

```
static var applicationIsNotAuthorized: AVError.Code
```

The app isn't authorized to play media.

```
static var autoWhiteBalanceNotLocked: AVError.Code
```

```
static var compositionTrackSegmentsNotContiguous: AVError.Code
```

The composition can't add the source media because it contains gaps.

```
static var contentIsNotAuthorized: AVError.Code
```

The user isn't authorized to play the media.

```
static var contentIsProtected: AVError.Code
```

The app isn't authorized to open the media.

```
static var contentIsUnavailable: AVError.Code
```

The captured content is unavailable.

```
static var contentKeyRequestCancelled: AVError.Code
```

The app canceled a request to retrieve a content key.

```
static var contentNotUpdated: AVError.Code
```

The system couldn't update the captured content.

```
static var createContentKeyRequestFailed: AVError.Code
```

The app couldn't create a content key request.

```
static var decodeFailed: AVError.Code
```

The system failed to decode the media.

```
static var decoderNotFound: AVError.Code
```

The system can't find a suitable decoder for the media.

```
static var decoderTemporarilyUnavailable: AVError.Code
```

A suitable decoder for the media is temporarily available.

```
static var deviceAlreadyUsedByAnotherSession: AVError.Code
```

Your app can't access the device because another session is currently using it.

```
static var deviceInUseByAnotherApplication: AVError.Code
```

Your app can't access the device because another app is currently using it.

`static var deviceLockedForConfigurationByAnotherProcess: AVError.Code`

Your app can't change device settings because another process currently controls the device.

`static var deviceNotConnected: AVError.Code`

Your app can't access the device because it isn't connected.

`static var deviceWasDisconnected: AVError.Code`

A previously connected device is no longer accessible.

`static var diskFull: AVError.Code`

Recording stopped because the disk is full.

`static var displayWasDisabled: AVError.Code`

Screen capture failed because the display was inactive.

`static var encodeFailed: AVError.Code`

The system couldn't encode the media data.

`static var encoderNotFound: AVError.Code`

The requested encoder isn't found.

`static var encoderTemporarilyUnavailable: AVError.Code`

An appropriate encoder isn't currently available.

`static var exportFailed: AVError.Code`

The requested export operation failed.

`static var externalPlaybackNotSupportedForAsset: AVError.Code`

The current asset doesn't support playback.

`static var failedToLoadMediaData: AVError.Code`

The system can't load the requested media data.

`static var failedToLoadSampleData: AVError.Code`

The system can't load the requested sample data.

`static var failedToParse: AVError.Code`

The system can't parse the media.

`static var fileAlreadyExists: AVError.Code`

A file with the same name exists at the location and you can't overwrite it.

`static var fileFailedToParse: ALError.Code`

The file is corrupt or in an unrecognized format.

`static var fileFormatNotRecognized: ALError.Code`

The system can't open the file because it's in an unrecognized format.

`static var fileTypeDoesNotSupportSampleReferences: ALError.Code`

The file type doesn't support sample references.

`static var followExternalSyncDeviceTimedOut: ALError.Code`

`static var formatUnsupported: ALError.Code`

The current asset format isn't supported.

`static var incompatibleAsset: ALError.Code`

You can't display the media because the device isn't capable of playing the content.

`static var incorrectlyConfigured: ALError.Code`

The system is incorrectly configured for the requested operation.

`static var invalidCompositionTrackSegmentDuration: ALError.Code`

You can't add the source media because its duration in the destination is invalid.

`static var invalidCompositionTrackSegmentSourceDuration: ALError.Code`

You can't add the source media because it has no duration.

`static var invalidCompositionTrackSegmentSourceStartTime: ALError.Code`

You can't add the source media because its start time in the destination is invalid.

`static var invalidOutputURLPathExtension: ALError.Code`

The path extension of the output URL is invalid.

`static var invalidSampleCursor: ALError.Code`

An invalid sample cursor produced an error.

`static var invalidSourceMedia: ALError.Code`

The system couldn't read the source media.

`static var invalidVideoComposition: ALError.Code`

You attempted to present an unsupported video composition.

`static var malformedDepth: ALError.Code`

The depth data isn't properly structured.

`static var maximumDurationReached: AVError.Code`

The recording stopped because it reached the file's maximum duration.

`static var maximumFileSizeReached: AVError.Code`

The recording stopped because it reached the file's maximum size.

`static var maximumNumberOfSamplesForFileFormatReached: AVError.Code`

The recording stopped because it reached the file's maximum number of samples.

`static var maximumStillImageCaptureRequestsExceeded: AVError.Code`

Your app can't take a photo because there are too many unfinished photo capture requests.

`static var mediaChanged: AVError.Code`

Recording stopped because the format of the source media changed.

`static var mediaDiscontinuity: AVError.Code`

Recording stopped because there was an interruption in the input media.

`static var mediaExtensionConflict: AVError.Code`

`static var mediaExtensionDisabled: AVError.Code`

`static var mediaServicesWereReset: AVError.Code`

The system couldn't perform the operation because media services were unavailable.

`static var noCompatibleAlternatesForExternalDisplay: AVError.Code`

The system found no compatible external displays.

`static var noDataCaptured: AVError.Code`

The recording failed because the system received no data.

`static var noImageAtTime: AVError.Code`

No image is available in the media at the indicated time.

`static var noLongerPlayable: AVError.Code`

The asset is no longer playable.

`static var noSmartFramingsEnabled: AVError.Code`

`static var noSourceTrack: AVError.Code`

The asset doesn't contain a source track.

`static var operationCancelled: AVError.Code`

The asset handled a request to cancel loading a property value asynchronously.

`static var operationInterrupted: ALError.Code`

An interruption occurred while performing a reading or writing operation.

`static var operationNotAllowed: ALError.Code`

The requested operation isn't allowed.

`static var operationNotSupportedForAsset: ALError.Code`

Your app attempted to perform an unsupported operation with the asset.

`static var operationNotSupportedForPreset: ALError.Code`

Your app attempted to perform an unsupported operation for the current preset.

`static var outOfMemory: ALError.Code`

The operation couldn't finish because there isn't enough memory available to process the media.

`static var recordingAlreadyInProgress: ALError.Code`

Your app attempted to start recording a movie file while an existing recording is underway.

`static var referenceForbiddenByReferencePolicy: ALError.Code`

The current reference restrictions prevent the system from loading referenced media.

`static var rosettaNotInstalled: ALError.Code`

The system doesn't have Rosetta installed and can't perform the requested operation.

`static var sandboxExtensionDenied: ALError.Code`

The system denied issuing the sandbox extension.

`static var screenCaptureFailed: ALError.Code`

An unexpected problem occurred that prevented screen capture.

`static var segmentStartedWithNonSyncSample: ALError.Code`

The operation attempted to write a new MPEG-4 segment that didn't start with a sync sample.

`static var serverIncorrectlyConfigured: ALError.Code`

The configuration of the HTTP server that streams the media resource isn't correct.

`static var sessionConfigurationChanged: ALError.Code`

Recording stopped because the configuration of media sources and destinations changed.

`static var sessionHardwareCostOverage: ALError.Code`

Your app requested too many camera hardware resources.

```
static var sessionNotRunning: ALError.Code
```

The recording couldn't start because the session isn't running.

```
static var sessionWasInterrupted: ALError.Code
```

The recording stopped because the system interrupted the audio session.

```
static var toneMappingFailed: ALError.Code
```

The requested tone mapping failed.

```
static var torchLevelUnavailable: ALError.Code
```

The specified torch level is valid but currently unavailable, possibly due to overheating.

```
static var undecodableMediaData: ALError.Code
```

The system couldn't decode the media data.

```
static var unknown: ALError.Code
```

An unknown error occurred.

```
static var unsupportedDeviceActiveFormat: ALError.Code
```

The capture session doesn't support the camera device's active format.

```
static var unsupportedOutputSettings: ALError.Code
```

Your app requested unsupported output settings.

```
static var videoCompositorFailed: ALError.Code
```

The compositor couldn't composite video frames.

```
static var deviceIsNotAvailableInBackground: ALError.Code
```

You attempted to start a capture session in the background, which isn't allowed in iOS.

Deprecated

Error properties

```
var device: AVCaptureDevice?
```

The capture device in use.

```
var device: String?
```

```
var fileSize: Int64?
```

The asset file size.

`var fileType: AVFileType?`

The asset file type.

`var mediaSubtypes: [Int]?`

An array of media subtypes.

`var mediaType: AVMediaType?`

The asset media type.

`var mediaType: String?`

The media type.

`var persistentTrackID: CMPersistentTrackID?`

The persistent track ID, if the track exists.

`var presentationTimeStamp: CMTime?`

The presentation time stamp.

`var processID: Int?`

The process ID.

`var recordingSuccessfullyFinished: Bool?`

A Boolean value that indicates whether recording finished successfully.

`var time: CMTime?`

The time duration of the error.

User info keys

`let ALErrorDeviceKey: String`

The user information key to retrieve the device name.

`let ALErrorDiscontinuityFlagsKey: String`

The user information key to retrieve discontinuity flags.

`let ALErrorFileSizeKey: String`

The user information key to retrieve the file size in bytes.

`let ALErrorFileTypeKey: String`

The user information key to retrieve the file type.

`let ALErrorMediaTypeKey: String`

The user information key to retrieve the media type.

```
let AVErrorMediaSubTypeKey: String
```

The user information key to retrieve the media subtype.

```
let AVErrorPersistentTrackIDKey: String
```

The user information key to retrieve the track's persistent identifier.

```
let AVErrorPIDKey: String
```

The user information key to retrieve the process ID value.

```
let AVErrorPresentationTimeStampKey: String
```

The user information key to retrieve the presentation time stamp.

```
let AVErrorRecordingSuccessfullyFinishedKey: String
```

The user information key to retrieve a Boolean value that indicates whether recording finished successfully.

```
let AVErrorTimeKey: String
```

The user information key to retrieve the error time.

Relationships

Conforms To

CustomNSError
Equatable
Error
Hashable
Sendable
SendableMetatype

See Also

Errors

```
let AVFoundationErrorDomain: String
```

The error domain of AVFoundation errors.