AVKit / Presenting Navigation Markers

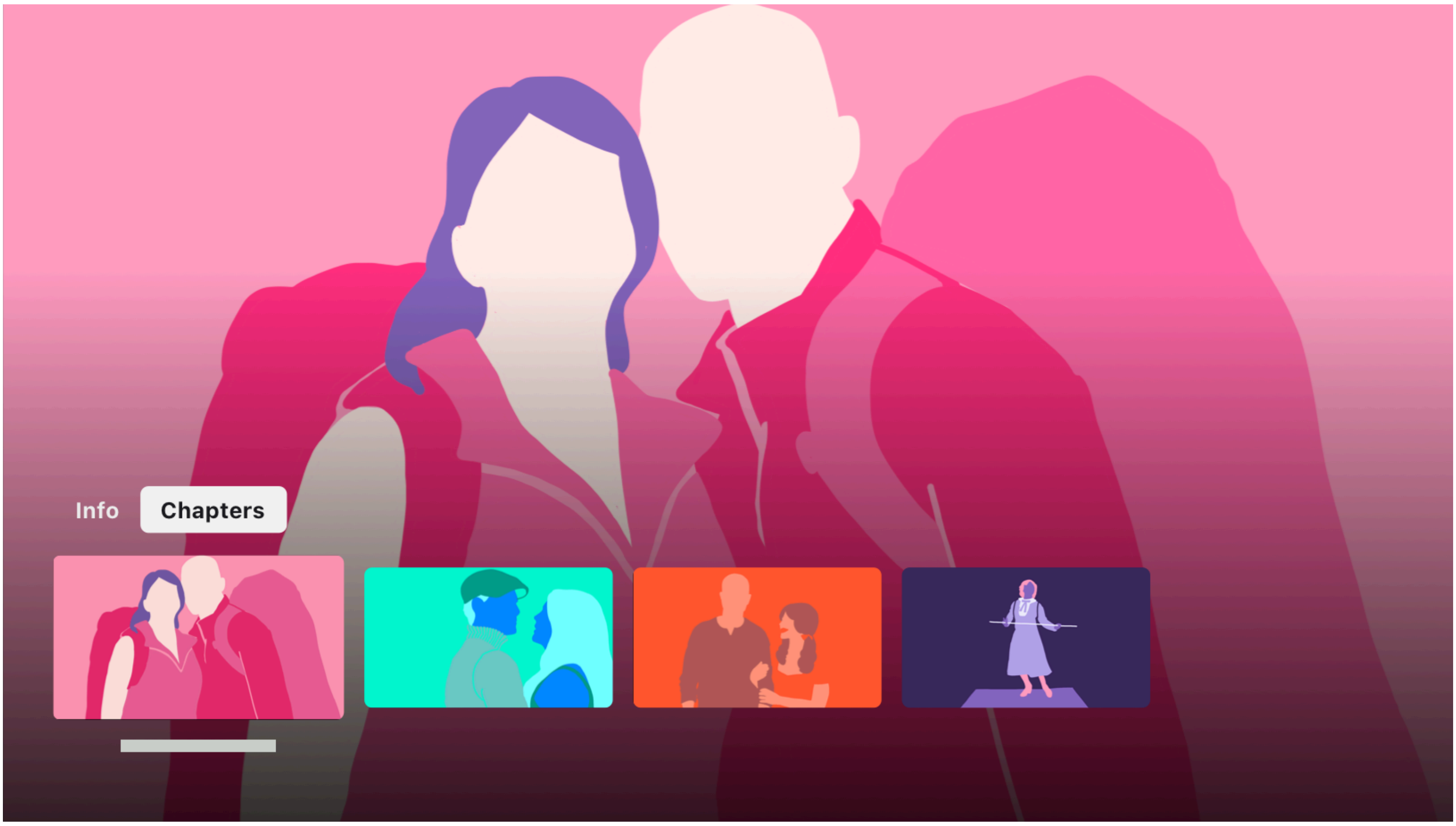
Article

Presenting Navigation Markers

Present navigation markers in the Chapters panel to help users quickly navigate your content.

Overview

To help users navigate your content, the Chapters panel presents navigation markers that represent points of interest within the media's timeline. Users can skip to desired content by selecting a marker with the Siri Remote.

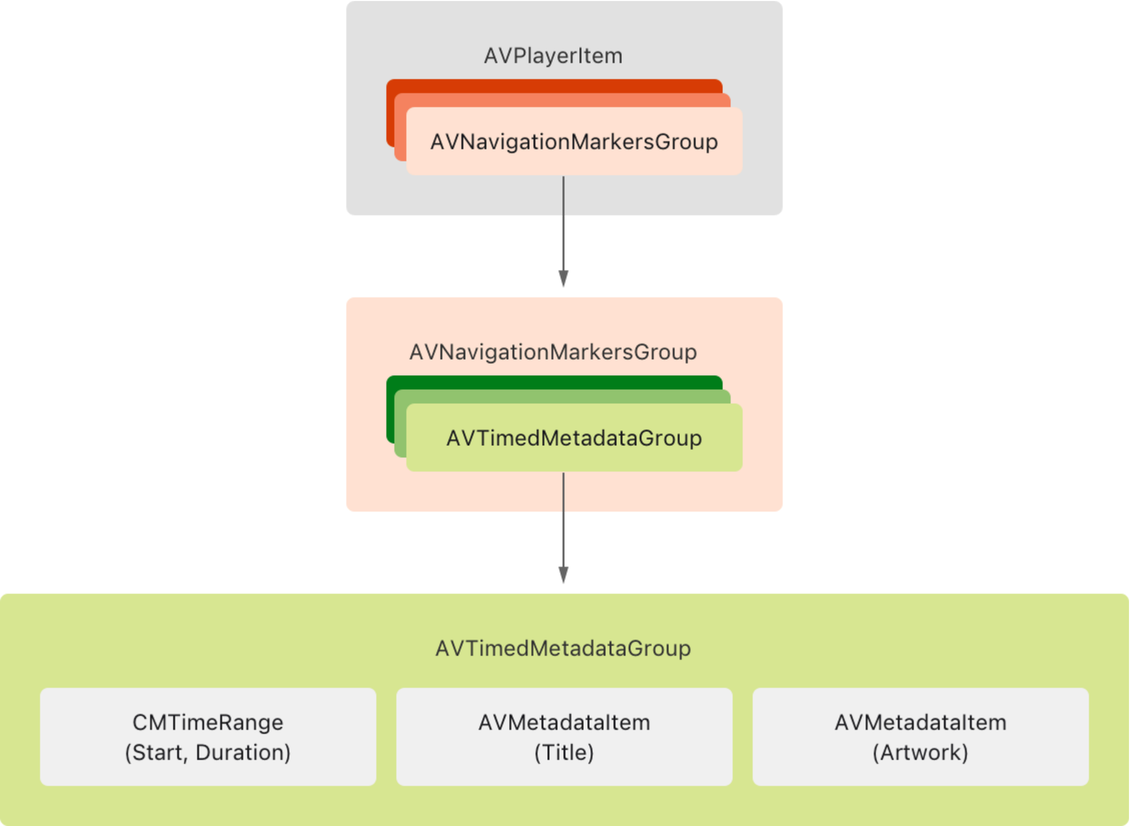


Set the Navigation Markers

In tvOS, a AVPlayerItem contains a navigationMarkerGroups property you use to supply chapter information. Set this property to an array of AVNavigationMarkersGroup objects to define the navigation markers for the current media.

> Note
>
> Although the player item defines the navigationMarkerGroups property as an array, the system only supports the first group in the array.

An AVNavigationMarkersGroup contains one or more AVTimedMetadataGroup objects, each representing an individual marker presented in the player's Info panel. Each AVTimedMetadataGroup stores a time range in the asset's timeline to which this marker applies, an array of AVMetadataItem objects to define the marker's title, and, optionally, its thumbnail artwork.



The following code shows how you can present a chapter list for your media:

```
struct Chapter {
    let title: String
    let imageName: String
    let startTime: TimeInterval
    let endTime: TimeInterval
}

...
```

```
func setupPlayback() {
    ...
    playerItem.navigationMarkerGroups = makeNavigationMarkerGroups()
    ...
}

private func makeNavigationMarkerGroups() -> [AVNavigationMarkersGroup] {

    let chapters = [
        Chapter(title: "Chapter 1", imageName: "chapter1", startTime: 0.0, endTime:
        Chapter(title: "Chapter 2", imageName: "chapter2", startTime: 60.0, endTime:
        Chapter(title: "Chapter 3", imageName: "chapter3", startTime: 120.0, endTime
        Chapter(title: "Chapter 4", imageName: "chapter4", startTime: 180.0, endTime
    ]

    var metadataGroups = [AVTimedMetadataGroup]()

    // Iterate over the defined chapters and build a timed metadata group object for
    chapters.forEach { chapter in
        metadataGroups.append(makeTimedMetadataGroup(for: chapter))
    }

    return [AVNavigationMarkersGroup(title: nil,
                                     timedNavigationMarkers: metadataGroups)]
}

private func makeTimedMetadataGroup(for chapter: Chapter) -> AVTimedMetadataGroup {
    var metadata = [AVMetadataItem]()

    // Create a metadata item that contains the chapter title.
    let titleItem = makeMetadataItem(.commonIdentifierTitle, value: chapter.title)
    metadata.append(titleItem)
    if let image = UIImage(named: chapter.imageName),
       let pngData = image.pngData() {
        // Create a metadata item that contains the chapter thumbnail.
        let imageItem = makeMetadataItem(.commonIdentifierArtwork, value: pngData)
        metadata.append(imageItem)
    }

    // Create a time range for the metadata group.
    let timescale: Int32 = 600
    let startTime = CMTime(seconds: chapter.startTime, preferredTimescale: timescale
    let endTime = CMTime(seconds: chapter.endTime, preferredTimescale: timescale)
    let timeRange = CMTimeRangeFromTimeToTime(start: startTime, end: endTime)

    return AVTimedMetadataGroup(items: metadata, timeRange: timeRange)
}

private func makeMetadataItem(_ identifier: AVMetadataIdentifier, value: Any) -> AVM
    let item = AVMutableMetadataItem()
    item.identifier = identifier
    item.value = value as? NSCopying & NSObjectProtocol
    item.extendedLanguageTag = "und"
    return item.copy() as! AVMetadataItem
}
```

See Also

tvOS playback and capture

Customizing the tvOS Playback Experience

Adopt the latest features of the redesigned tvOS player user interface to provide a more streamlined way to watch your content.

Working with Interstitial Content

Present additional content alongside your main media presentation using HTTP Live Streaming support.

Presenting Content Proposals in tvOS

Display a preview of an upcoming media item at the conclusion of the currently playing media item.

Working with Overlays and Parental Controls in tvOS

Add interactive overlays, parental controls, and livestream channel flipping using a player view controller.

Supporting Continuity Camera in your tvOS app

Capture high-quality photos, video, and audio in your Apple TV app by connecting an iPhone or iPad as a continuity device.

`class AVPlayerViewController`

A view controller that displays content from a player and presents a native user interface to control playback.

`protocol AVPlayerViewControllerDelegate`

A protocol that defines the methods to implement to respond to player view controller events.

`class AVInterstitialTimeRange`

A time range in an audiovisual presentation for content with an interstitial designation, such as advertisements or legal notices.

`class AVNavigationMarkersGroup`

A set of markers for navigating playback of an audiovisual presentation.

`class AVContentProposalViewController`

A view controller that proposes content to watch next.

`class AVDisplayManager`

A tvOS management object that controls whether a TV switches modes to match the video's native mode.

`class AVContinuityDevicePickerViewController`

A view controller that provides an interface to a person so they can select and connect a continuity device to the system.

`protocol AVContinuityDevicePickerViewControllerDelegate`

An interface that responds to events from a continuity device picker view controller.