

Media Playback and The Top Shelf

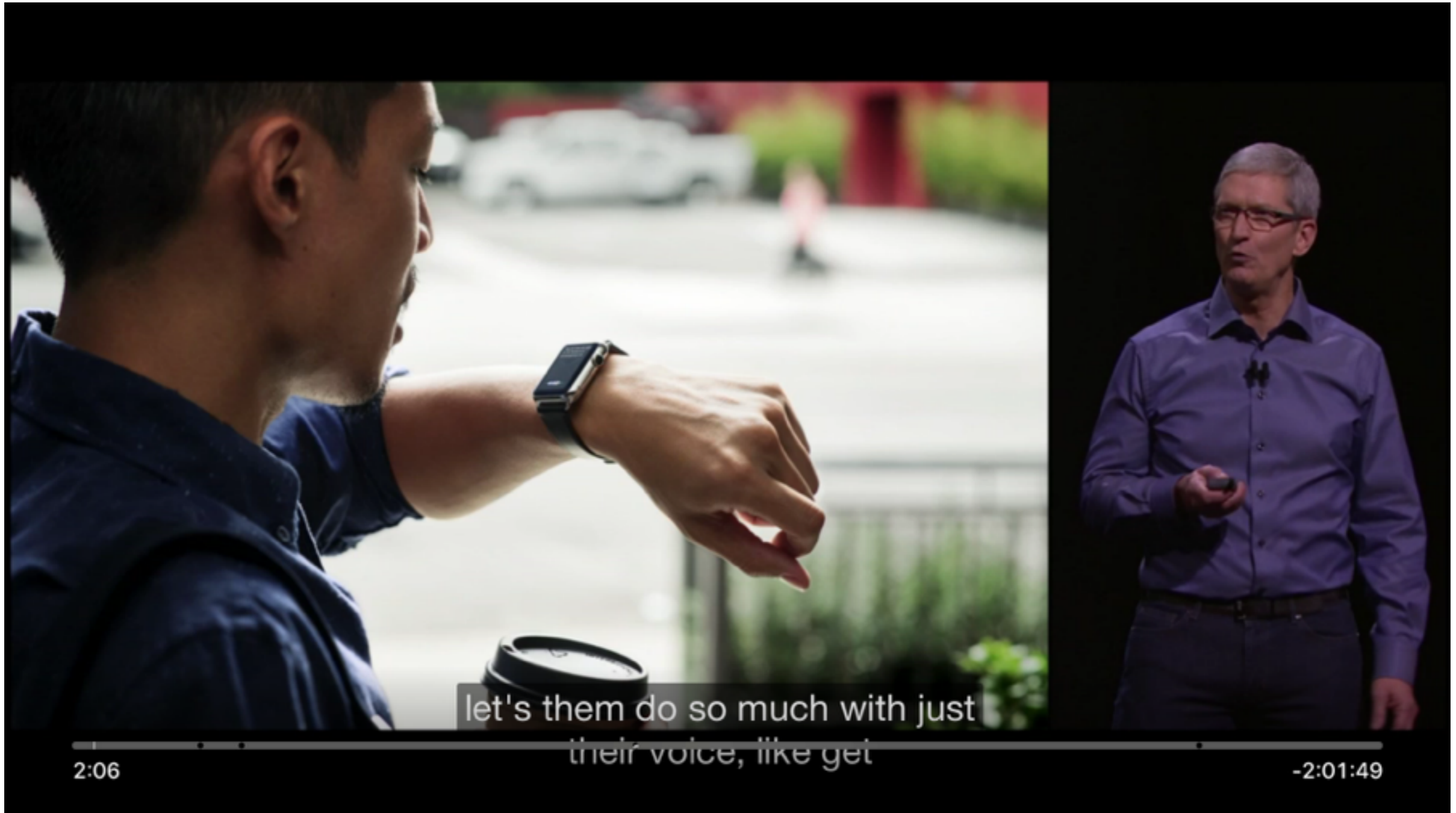
CS193W - Spring 2016 - Lecture 8

Today

- Playing Videos on Apple TV
- The Top Shelf

Playing Videos

AVPlayerViewController



Swiping Down Reveals Metadata and Options

Info

Subtitles

Audio

Apple Special Event, September 2015

2 hr 4 min

Check out iPhone 6s and iPhone 6s Plus, learn about the powerful iPad Pro, take a look at the new features and bands for Apple Watch, and see the premiere of the all-new Apple TV.

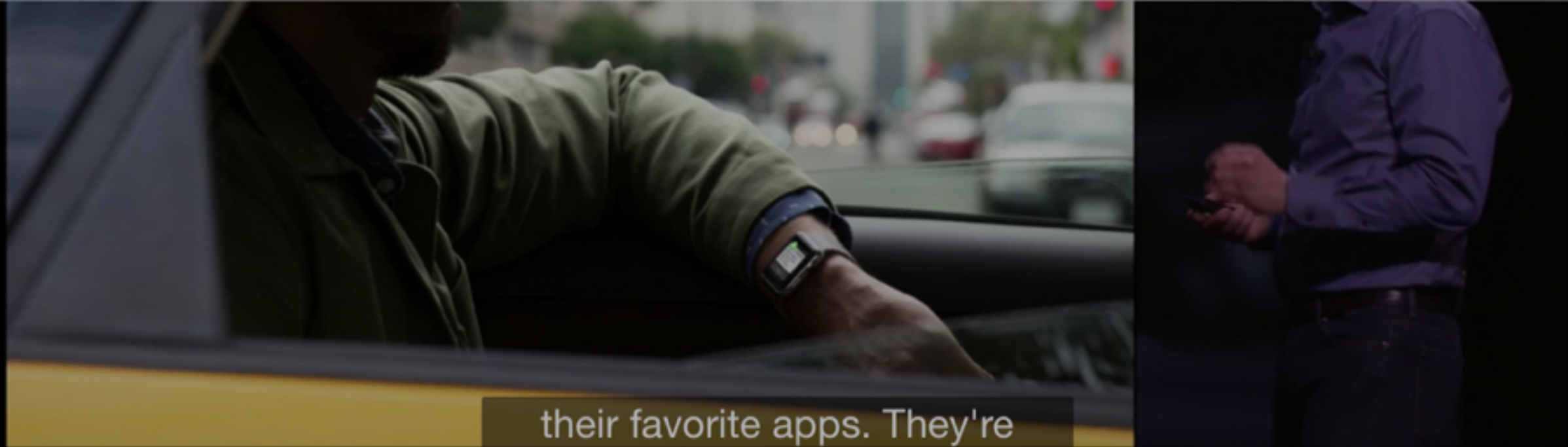
Apple Watch

iPad Pro

Apple Pencil

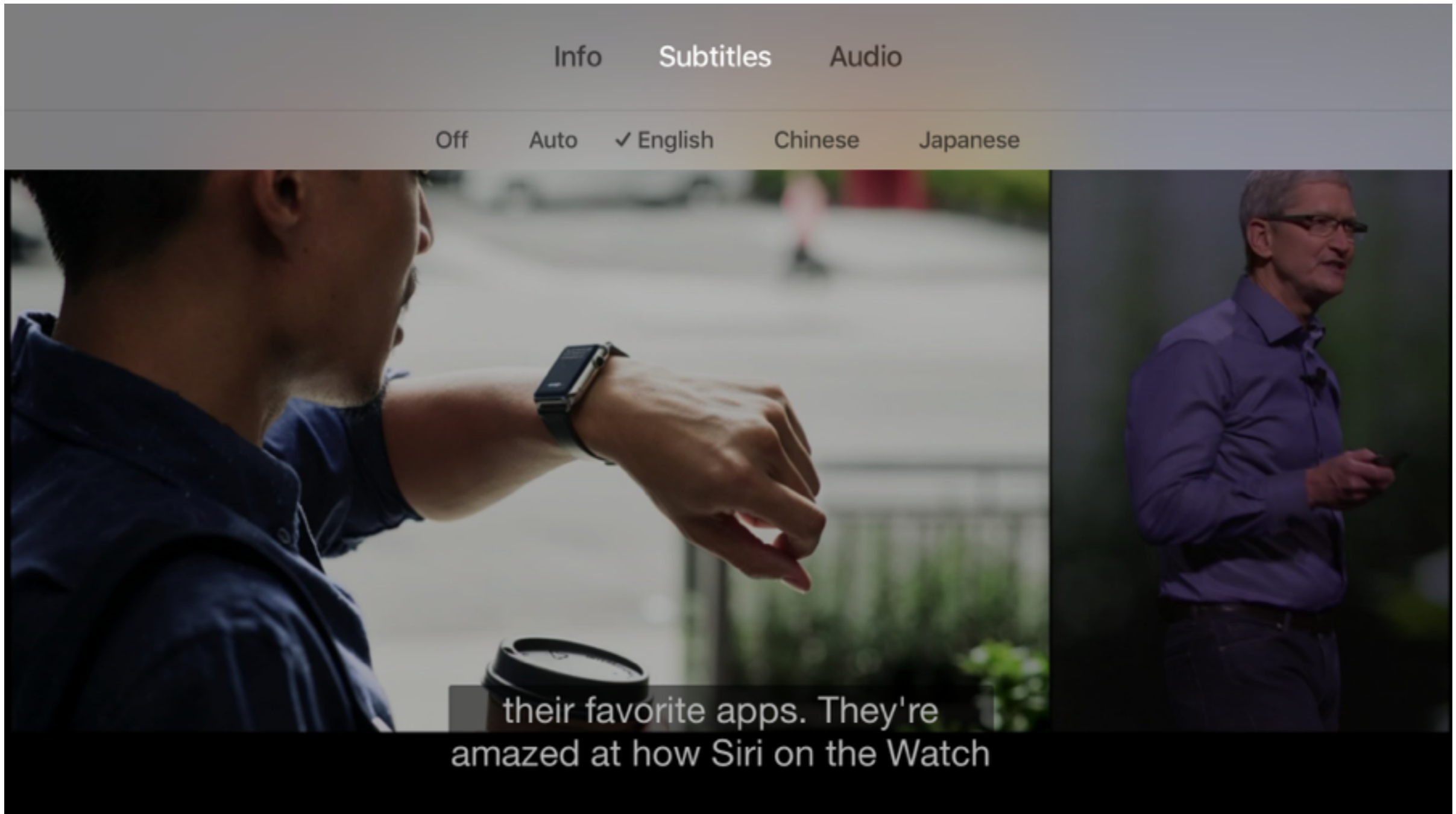
Apple TV

iPhone



their favorite apps. They're
amazed at how Siri on the Watch

Subtitles




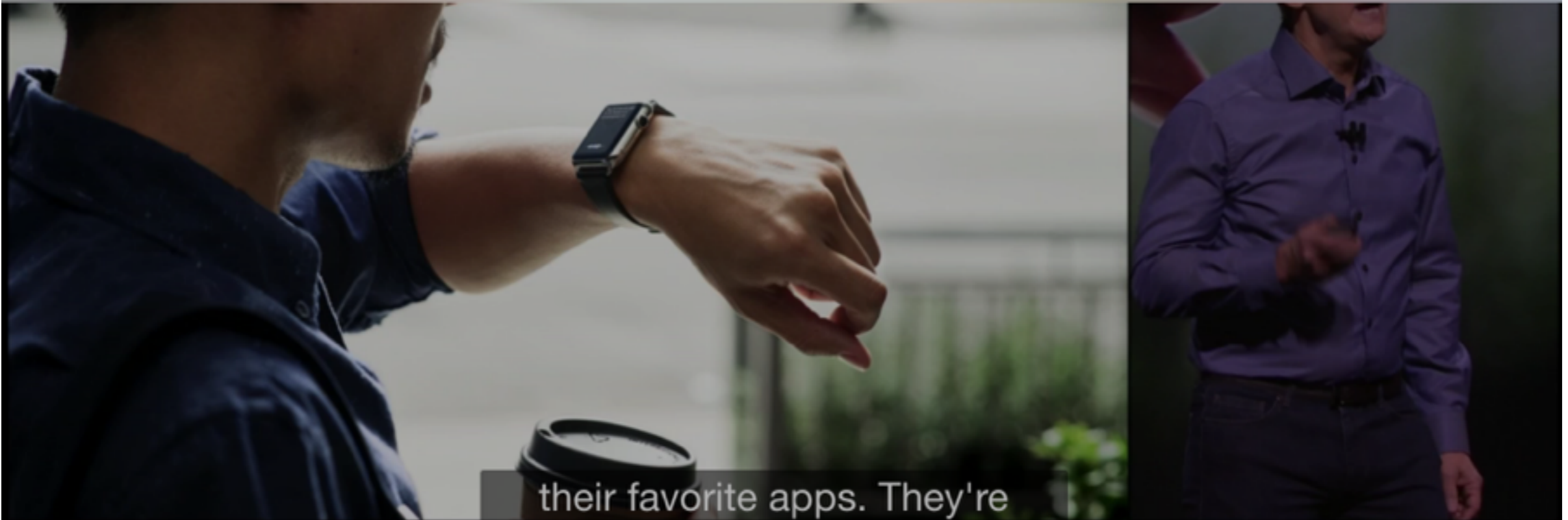
Audio

Info Subtitles Audio

LANGUAGE
✓ Unknown

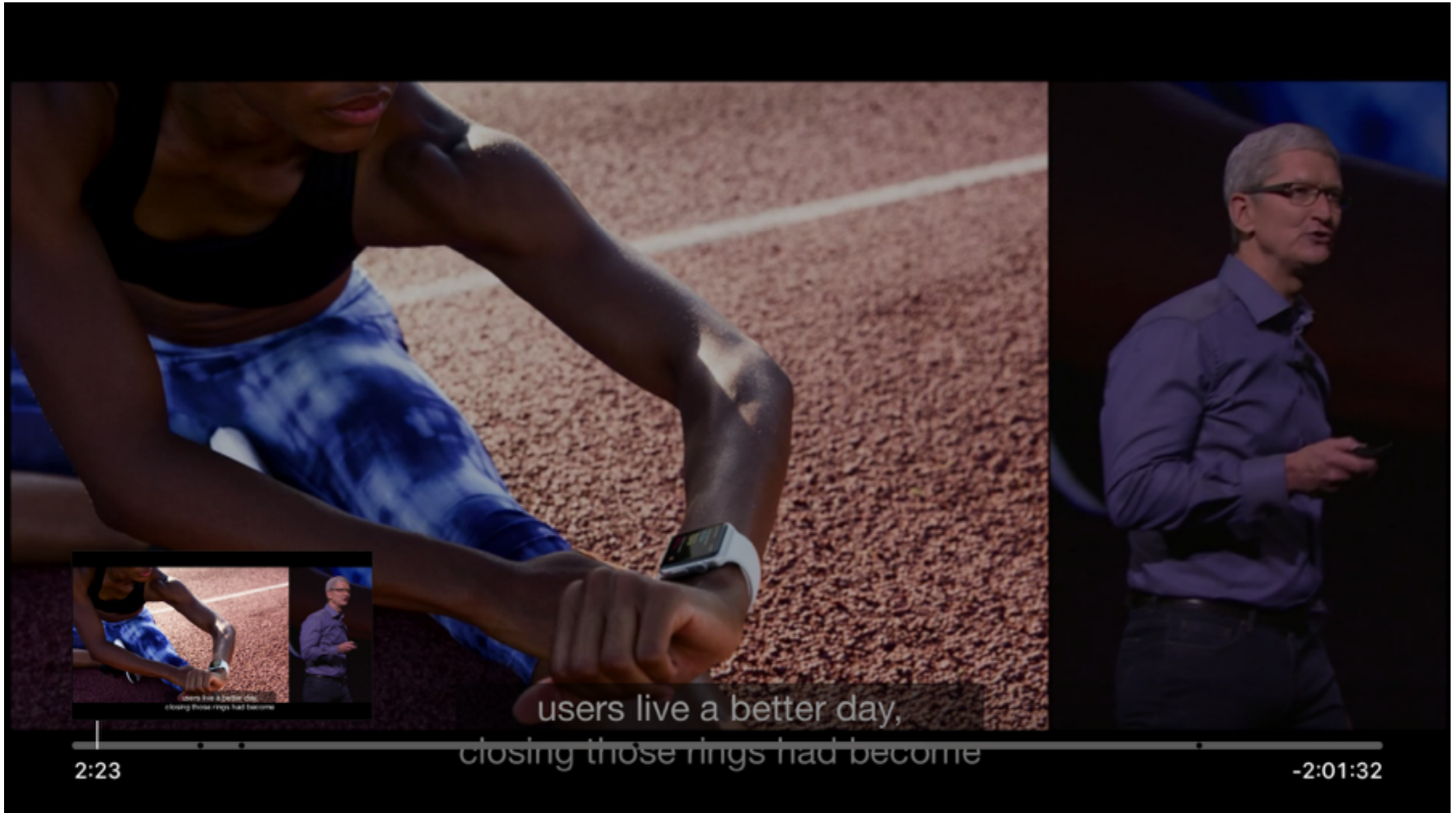
SOUND
✓ Full Dynamic Range
Reduce Loud Sounds

SPEAKER
✓  Mike's MacBook F



their favorite apps. They're amazed at how Siri on the Watch

Trick Play



Playing Media with AVPlayerViewController

AVAsset

A media file or stream

AVPlayerItem

Represents the state of an **AVAsset** as it is playing

AVPlayer

Controls playback of a media item

AVPlayerViewController

Displays videos using an **AVPlayer**

AVPlayer Methods

rate

The speed at which to play the media. 1 is regular speed, .5 is half speed, -1 is reversed, etc.

play()

Set the rate to 1

pause()

Set the rate to 0

seekToTime(_ time: CMTime)

**seekToTime(_ time: CMTime,
toleranceBefore toleranceBefore: CMTime,
toleranceAfter toleranceAfter: CMTime,
completionHandler completionHandler: (Bool) -> Void)**

Using a AVPlayerViewController

```
let mediaURL = NSURL(string: "https://www.example.com/example.m3u8")
if let mediaURL = mediaURL {
    let asset = AVURLAsset(URL: mediaURL)
    let playerItem = AVPlayerItem(asset: asset)
    let player = AVPlayer(playerItem: playerItem)
    let playerViewController = AVPlayerViewController()
    playerViewController.player = player
    presentViewController(playerViewController, animated: true, completion: {
        player.play()
    })
}
```

What's a .m3u8 file?

- A playlist file used for pointing media players at media sources
- Can specify a playlist of media
- Sources can be local or over the Internet

Specifying Streams of Various Quality

- Depending on the quality of the Internet connection available, AVPlayer will dynamically use different bitrates
- The M3U8 file specifies all the available streams and their bitrates
- It also can specify audio streams of differing qualities, close captioning in different languages, metadata, and images used for Trick Play scrubbing

M3U8 Playlist

#EXTM3U

#EXT-X-VERSION:4

#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="subs",NAME="English",DEFAULT=YES,AUTOSELECT=YES,FORCED=NO,LANGUAGE="eng",URI="http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/cc/eng.m3u8"

#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="subs",NAME="Chinese",DEFAULT=NO,AUTOSELECT=NO,FORCED=NO,LANGUAGE="zho",URI="http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/cc/chn.m3u8"

#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="subs",NAME="Japanese",DEFAULT=NO,AUTOSELECT=NO,FORCED=NO,LANGUAGE="jpn",URI="http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/cc/jpn.m3u8"

#EXT-X-STREAM-INF:BANDWIDTH=1423866,CODECS="avc1.77.30,mp4a.40.2",RESOLUTION=640x360,SUBTITLES="subs"

http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/1200/1200_vod.m3u8

#EXT-X-I-FRAME-STREAM-INF:BANDWIDTH=711933,CODECS="avc1.77.30,mp4a.40.2",RESOLUTION=640x360,URI="http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/1200/1200_vod_I-Frame.m3u8"

#EXT-X-STREAM-INF:BANDWIDTH=8840946,CODECS="avc1.580028,mp4a.40.2",RESOLUTION=1920x1080,SUBTITLES="subs"

http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/8500/8500_vod.m3u8

#EXT-X-I-FRAME-STREAM-INF:BANDWIDTH=4420473,CODECS="avc1.580028,mp4a.40.2",RESOLUTION=1920x1080,URI="http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijklkb/vod2/8500/8500_vod_I-Frame.m3u8"

#EXT-X-STREAM-INF:BANDWIDTH=6623635,CODECS="avc1.58001f,mp4a.40.2",RESOLUTION=1280x720,SUBTITLES="subs"

[http://p.events-delivery.apple.com.edgesuite.net/
1509pijnedfvopihbefvpijkljb/
vod2/8500/8500_vod_I-Frame.m3u8](http://p.events-delivery.apple.com.edgesuite.net/1509pijnedfvopihbefvpijkljb/vod2/8500/8500_vod_I-Frame.m3u8)

#EXTM3U

#EXT-X-VERSION:4

#EXT-X-TARGETDURATION:13

#EXT-X-MEDIA-SEQUENCE:1

#EXT-X-I-FRAMES-ONLY

#EXTINF:3.003000,

#EXT-X-BYTERANGE:75952@564

8500_vod_00001.ts

#EXTINF:3.003000,

#EXT-X-BYTERANGE:76140@3130388

8500_vod_00001.ts


#EXTINF:3.003000,

#EXT-X-BYTERANGE:74072@6261340

8500_vod_00001.ts

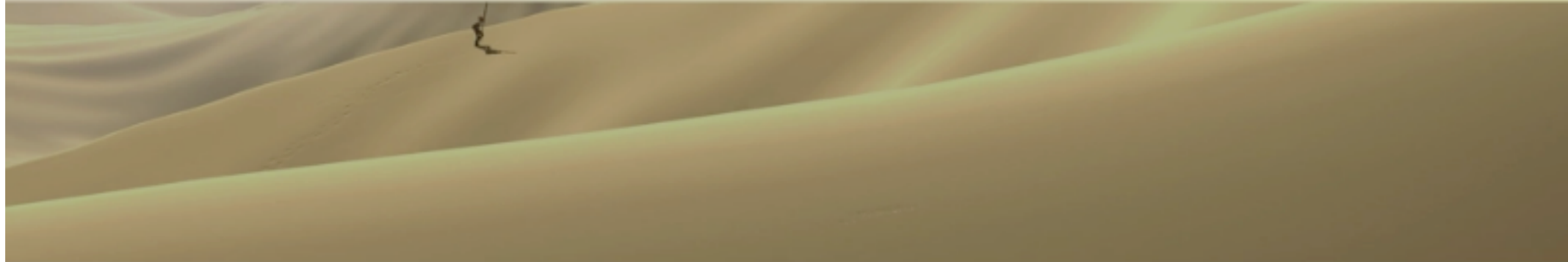
Metadata

Info Subtitles Audio



Sintel
15 min Fantasy **PG-13**
A wandering warrior finds an unlikely friend in the form of a young dragon. The two develop a close bond, until one day the dragon is snatched away. She then sets out on a relentless quest to reclaim her friend, finding in the end that her quest exacts a far greater price than she had ever imagined.

Chapter 3 Chapter 4 Chapter 5 Chapter 6 ✓ Chapter 7 Chapter 8



AVMetadataItem

identifier

There are many possible identifiers but the ones shown by `AVPlayerViewController` are

`AVMetadataCommonIdentifierTitle`
`AVMetadataCommonIdentifierDescription`
`AVMetadataIdentifieriTunesMetadataContentRating`
`AVMetadataIdentifierQuickTimeMetadataGenre`
`AVMetadataCommonIdentifierArtwork`

value

The value of the metadata item

extendedLanguageTag

The IETF BCP 47 language tag associated with the option, or “und” if undetermined

externalMetadata

`AVPlayerItem.externalMetadata` takes an array of `AVMetadataItem`

```
let titleItem = AVMutableMetadataItem()  
titleItem.identifier = AVMetadataCommonIdentifierTitle  
titleItem.value = "Apple Special Event, September 2015"  
titleItem.extendedLanguageTag = "und"
```

```
let playerItem = AVPlayerItem(asset: asset)  
playerItem.externalMetadata = [titleItem]
```

Navigation Markers

Info

Subtitles

Audio

Apple Special Event, September 2015

2 hr 4 min

Check out iPhone 6s and iPhone 6s Plus, learn about the powerful iPad Pro, take a look at the new features and bands for Apple Watch, and see the premiere of the all-new Apple TV.

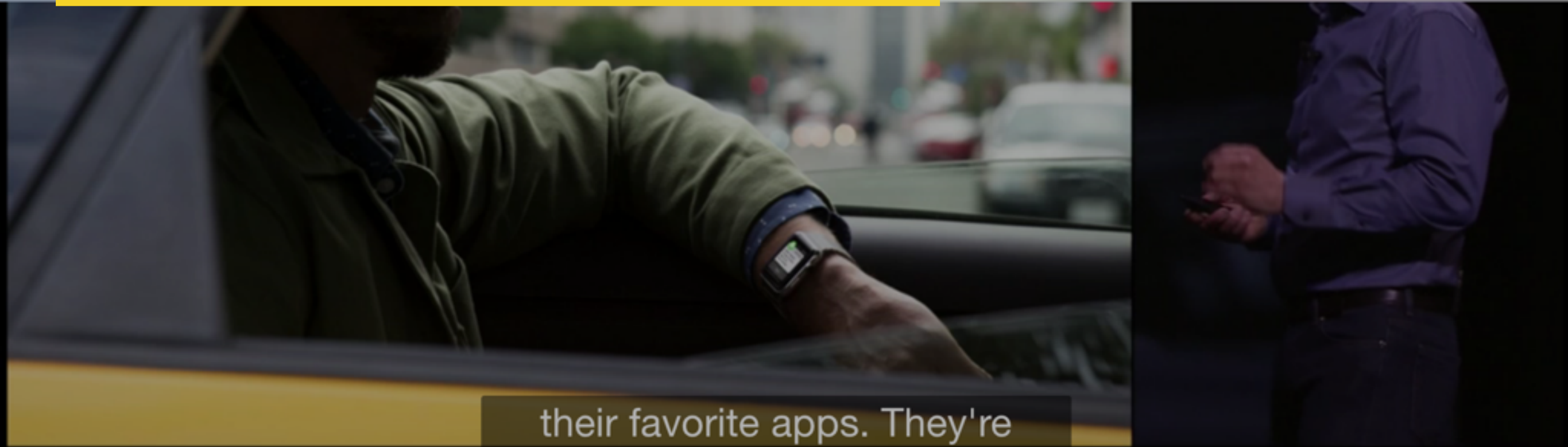
Apple Watch

iPad Pro

Apple Pencil

Apple TV


iPhone




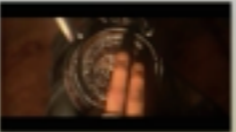





their favorite apps. They're
amazed at how Siri on the Watch

Navigation Markers With Images

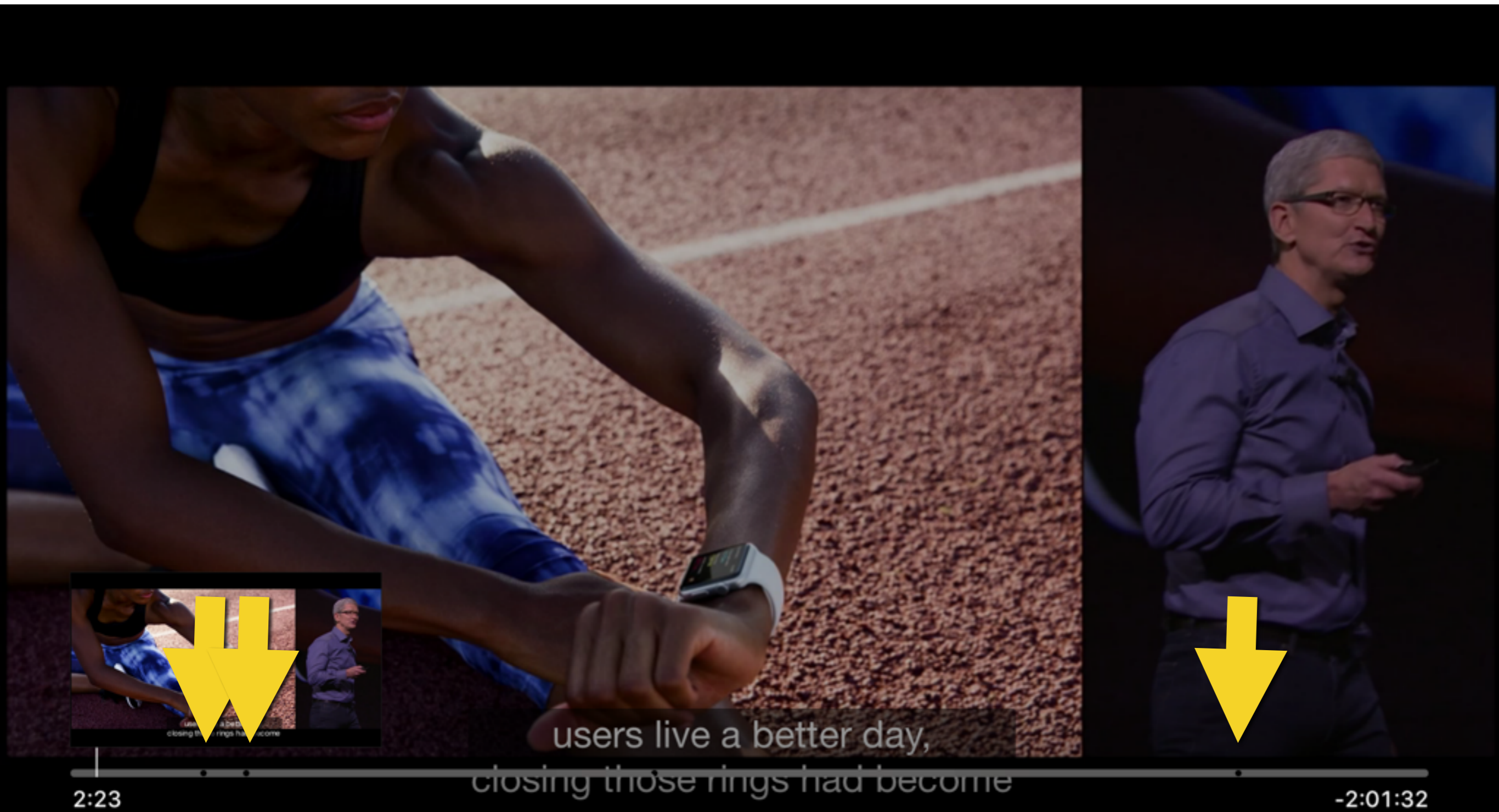
Info Subtitles Audio

 **Sintel**
15 min Fantasy **PG-13**
A wandering warrior finds an unlikely friend in the form of a young dragon. The two develop a close bond, until one day the dragon is snatched away. She then sets out on a relentless quest to reclaim her friend, finding in the end that her quest exacts a far greater price than she had ever imagined.

 Chapter 3  Chapter 4  Chapter 5  Chapter 6  ✓ Chapter 7  Chapter 8



Navigation Markers shown in Timeline



Classes Involved in Making Navigation Markers

CMTIMEVALUE

An Int64

CMTIMESCALE

An Int32

CMTIME

Pairs a time (in seconds) with a timescale. $\text{value}/\text{scale} = \text{seconds}$.

CMTIMERANGE

(start time, end time) pair

AVTIMEDMETADATAGROUP

Pairs a metadata item with a time range. Metadata items can have titles and thumbnails (artwork).

AVNAVIGATIONMARKERSGROUP

Pairs a title to a list of timed metadata groups

Specifying Navigation Groups for a Player Item

```
let cmStartTime = CMTimeMakeWithSeconds(Float64(550), 100)
let cmDuration = CMTimeMakeWithSeconds(Float64(60000), 100)
let timeRange = CMTimeRange(start: cmStartTime, duration: cmDuration) // 10 minutes starting at 5.5 seconds

// Create an `AVMetadataItem` for the title.
let titleItem = AVMutableMetadataItem()
titleItem.identifier = AVMetadataCommonIdentifierTitle
titleItem.value = "Apple Watch"
titleItem.extendedLanguageTag = "und"

// Create a `CMTimeRange` from the supplied information.
let timeRange = timeRangeFrom(startTime, to: endTime)

let timedMetadataGroup = AVTimedMetadataGroup(items: [titleItem], timeRange: timeRange)

let navGroup = AVNavigationMarkersGroup(title: "Announcements", timedNavigationMarkers: [timedMetadataGroup])

let playerItem = AVPlayerItem(asset: asset)
playerItem.navigationMarkerGroups = [navGroup]
```

Interstitials

- An *interstitial* is a piece of content that is part of the video but which does not show up in the timeline of the video and whose time does not count towards the total time shown in the timeline
- Useful for:
 - Ads
 - Content warnings
 - Legal notices

Interstitials Shown as Dots in the Timeline



Detecting Seeks

Implement AVPlayerViewControllerDelegate's

```
optional func playerViewController(_ playerViewController: AVPlayerViewController,  
willResumePlaybackAfterUserNavigatedFromTime oldTime: CMTime,  
toTime targetTime: CMTime)
```

This method fires only for complete, user-initiated navigation events. For example, if the user begins continuously scrubbing through the playback timeline, pausing at several times along the way before resuming playback, the player view controller calls this method only once.

Making interstitials non-skippable

`AVPlayerViewControllerDelegate` has callbacks:

```
playerViewController:willPresentInterstitialTimeRange:  
playerViewController:didPresentInterstitialTimeRange:
```

You can use these callbacks to disable seeking and scrubbing by setting `AVPlayerViewController`'s `requiresLinearPlayback`

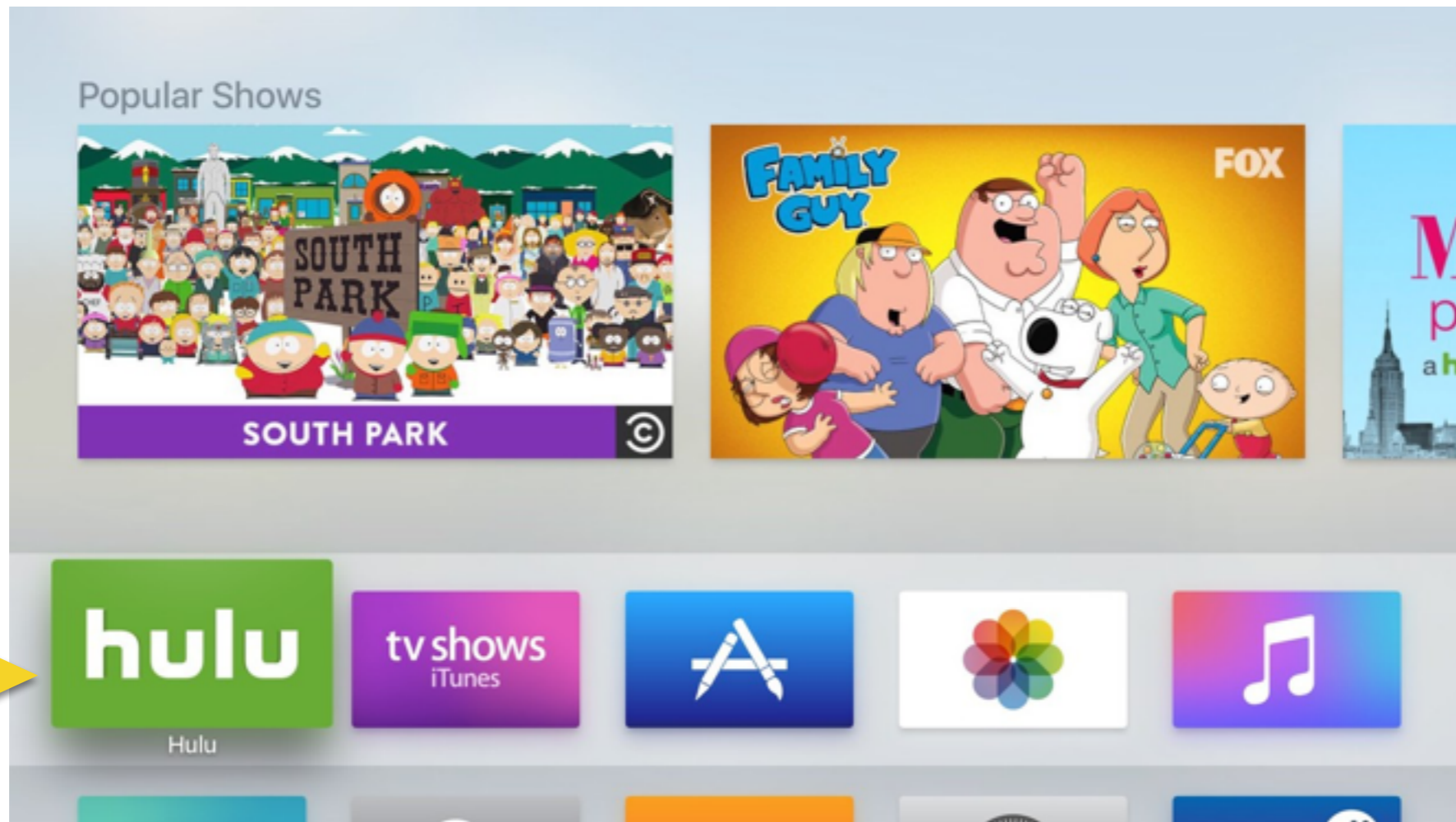
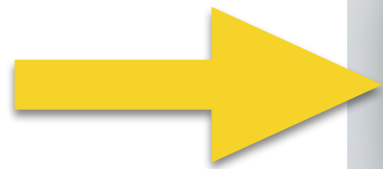
Use cases for skipping

1. The first time the user gets a particular interstitial, he is forced to watch it. If the user rewinds, the interstitial is not shown or is skippable.
2. Whenever the user rewinds or skip ahead past an ad interstitial, the ad is shown immediately, then the content is shown. This can be achieved by skipping to the interstitial, then skipping back to the current time after the interstitial finishes.

The Top Shelf

The Top Shelf

Currently Selected App



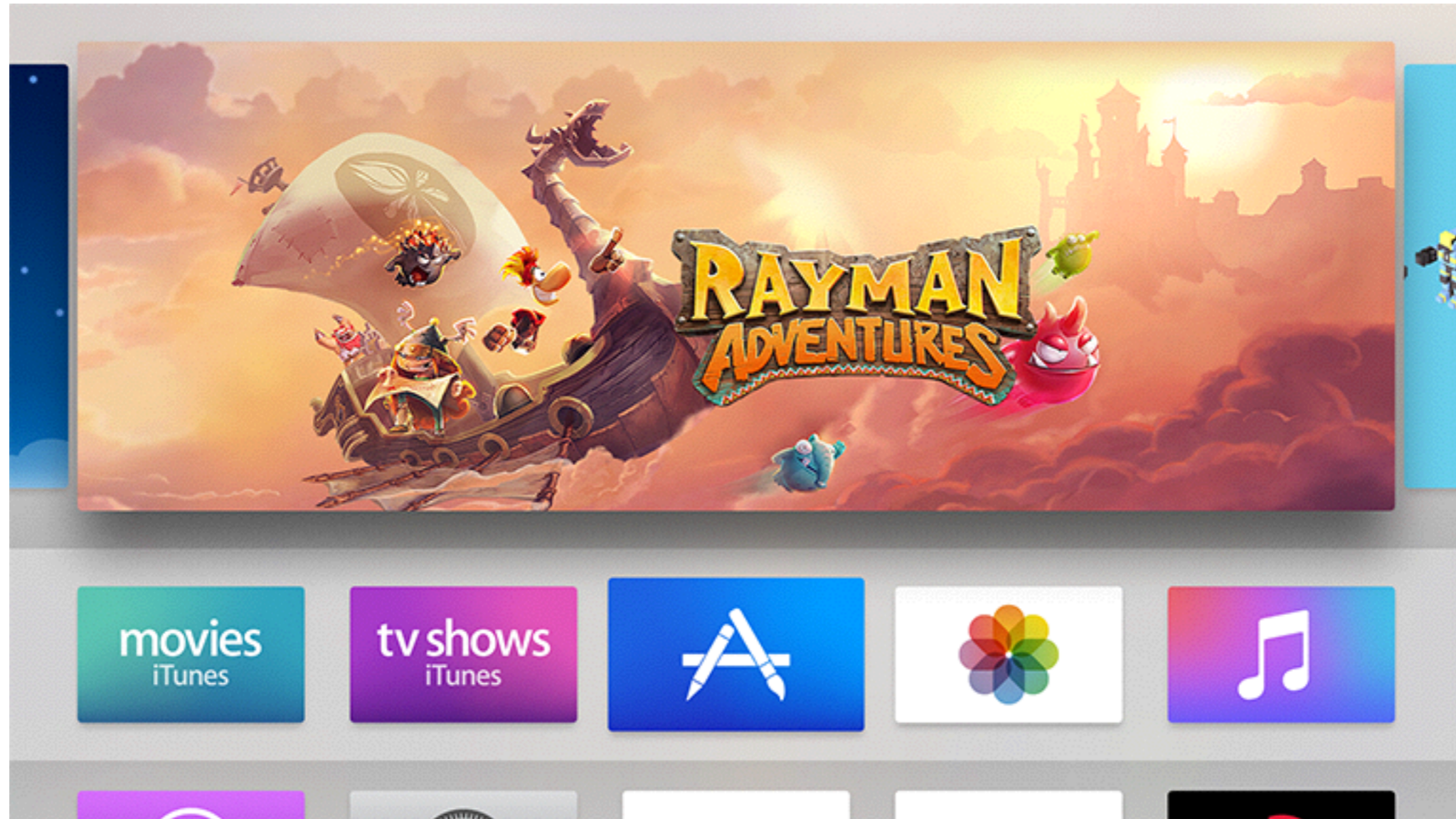
Three Kinds of Top Shelves

- Static Image (must be provided by all apps)
- Dynamic
 - Inset
 - Sectioned

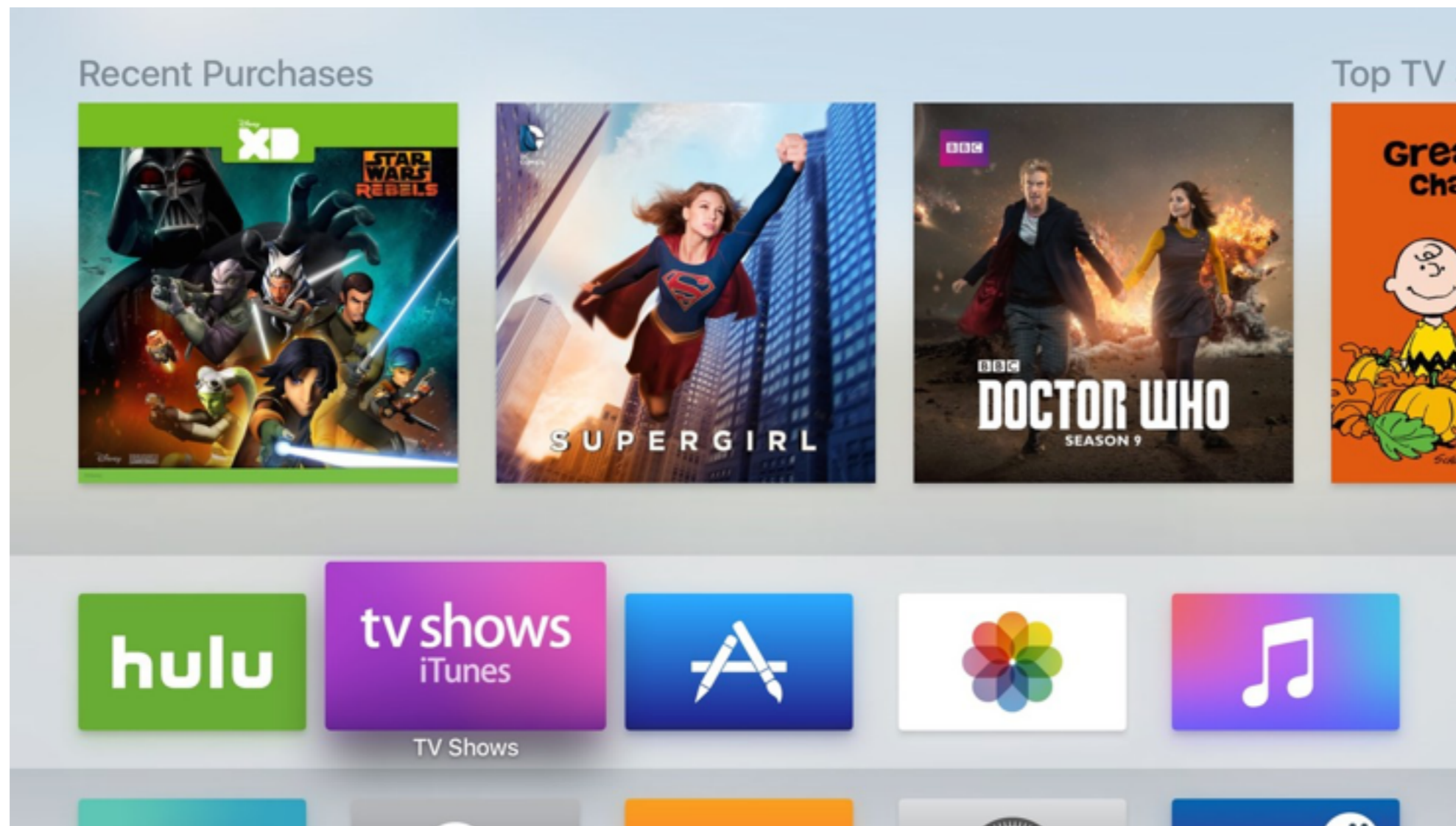
Static Image



Inset

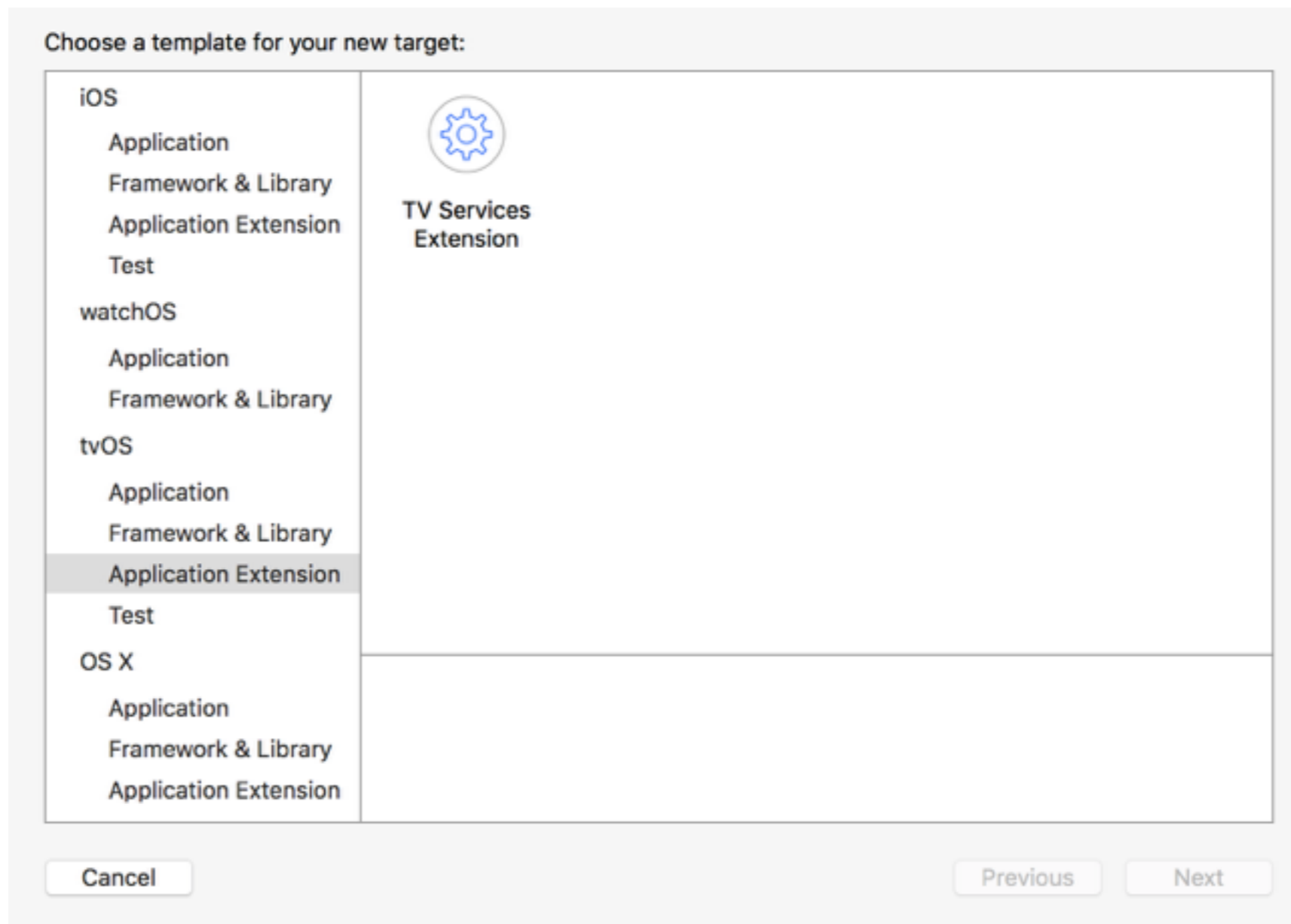


Sectioned



Adding a Top Shelf

File -> New -> Target...



TVTopShelfProvider

- An object that implements `TVTopShelfProvider` will be created
- You will need to return `.Inset` or `.Sectioned` for the `topShelfStyle` getter
- You will also need to return an array of `TVContentItems` to populate your Top Shelf

TVContentIdentifier

- Every piece of content must have a unique identifier
- Unique within an app, for *all time*
- Identifiers can have hierarchical structure
 - e.g. Series / Season / Episode
 - The structure is represented by *containers*

```
public init?(identifier: String, container: TVContentIdentifier?)
```

TVContentItem

```
public init?(contentIdentifier ident: TVContentIdentifier)
```

```
var imageURL: NSURL?
```

```
public var imageShape: TVContentItemImageShape
```

For `.Inset`, must be `.ExtraWide` (1940px by 692px)

For `.Section`, must be `.Poster` (404px by 608px), `.Square` (608px by 608px), or `.HDTV` (908px by 512px)

```
var title: String?
```

In the case of a `.Section`, the title of the Section. Otherwise the title is shown below the focused item.

```
var topShelfItems: [TVContentItem]?
```

In the case of a `.Section`, only the title is relevant; the actual items are nested in this property.

```
var playURL: NSURL?
```

The URL to be opened if the user presses the Play button while this item is in focus.

```
var displayURL: NSURL?
```

The URL to be opened if the user clicks while this item is in focus.

Opening Your App via a URL

- Each app can be associated with one or more custom URL schemes
- myapp://play/scrubs_season1_ep4
- myapp is the custom URL scheme
- You can specify this in your Info.plist

▼ URL types	⬆️ ⬆️ + -	Array	(1 item)
▼ Item 0		Dictionary	(1 item)
▼ URL Schemes	⬆️	Array	(5 items)
Item 0		String	myapp

Handling URL Opens

- In your application delegate, implement `application:openURL:options`
- Return true if you are able to handle the URL

Updating Top Shelf Content

- Content is cached. To update your Top Shelf content, post a `TVTopShelfItemsDidChangeNotification` notification