App Development Using TVMLKit

Part 1
Session 212

Nurinder Manj tvOS Engineer
Agenda

What is TVMLKit?
Agenda

What is TVMLKit?
Building an app with TVMLKit
Agenda

What is TVMLKit?
Building an app with TVMLKit
New features
Find Your Favorite Video Apps

New Games We Love
- Pixel Cup Soccer 16 $2.99
- MMX Hill Climb — Off Road
- Ball Escape!
- Sleepy Walker $0.99
- Icycle: On Thin Ice $2.99

What to Watch
- ENow
- CBS
- SyfyNow
- WATCH
- FOX NEWS
Disney presents a heartwarming comedy-adventure set in the modern mammal metropolis of Zootopia. With habitat neighborhoods like ritzy Sahara Square and frigid Tundratown, it's a melting pot where animals from every environment live together—a place where no matter...
Another Eternity
Purity Ring
Electronic 10 Songs 2015

When the going gets tough, the professionals turn pop. Or at least that's what the Canadian experimental/electronic pop duo Purity Ring have done on their second album, An...

1. Heartsigh - 3:19
2. Bodyache - 2:53
3. Push Pull - 3:27
4. Repetition - 3:38
5. Stranger Than Earth - 4:18
6. Begin Again - 3:37
7. Dust Hymn - 3:30
TVMLKit
What is TVMLKit?

Native experience
What is TVMLKit?

Native experience
XML and JavaScript
What is TVMLKit?

Native experience
XML and JavaScript
Configurable
What is TVMLKit?

Native experience
XML and JavaScript
Configurable
Extensible
What is TVMLKit?

Native experience
XML and JavaScript
Configurable
Extensible
Not a web browser
Building an App With TVMLKit
Building an App With TVMLKit

Xcode
TVML and Styles
Building an App With TVMLKit

Xcode
TVML and Styles
TVMLKit JS
Application Architecture

TVMLKit
Application Architecture
Application Architecture

TVMLKit

TVMLKit JS

Storage

DOM

XMLHttpRequest

App

MediaItem

NavigationDocument

Playlist

Settings

Player
TVML Application Template
Xcode window
class AppDelegate: ..., TVApplicationControllerDelegate {
    
    static let tvBaseURL = "http://localhost:9001/"
    static let tvBootURL = "\(AppDelegate.tvBaseURL)/application.js"
    
    func application(..., didFinishLaunchingWithOptions launchOptions: ...) -> Bool {
        
        let appControllerContext = TVApplicationControllerContext()
        if let javaScriptURL = NSURL(string: AppDelegate.tvBootURL) {
            appControllerContext.javaScriptApplicationURL = javaScriptURL
        }
        
        appController = TVApplicationController(context: appControllerContext, window: window, delegate: self)
        
        return true
    }
}
// Bootstrapping TVMLKit App

class AppDelegate: ..., TVApplicationControllerDelegate {

  ...

  static let tvBaseURL = "http://localhost:9001/

  static let tvBootURL = "\(AppDelegate.tvBaseURL)/application.js"

  func application(..., didFinishLaunchingWithOptions launchOptions: ...) -> Bool {

    ...

    let appControllerContext = TVApplicationControllerContext()

    if let javascriptURL = NSURL(string: AppDelegate.tvBootURL) {
      appControllerContext.javaScriptApplicationURL = javascriptURL
    }

    ...

    appController = TVApplicationController(context: appControllerContext, window: window,
                                          delegate: self)

    ...

    return true

  }

}
class AppDelegate: ..., TVApplicationControllerDelegate {
    ...
    static let tvBaseURL = "http://localhost:9001/
    static let tvBootURL = "\(AppDelegate.tvBaseURL)/application.js"
    func application(..., didFinishLaunchingWithOptions launchOptions: ...) -> Bool {
        ...
        let appControllerContext = TVApplicationControllerContext()
        if let javaScriptURL = NSURL(string: AppDelegate.tvBootURL) {
            appControllerContext.javaScriptApplicationURL = javaScriptURL
        }
        ...
        appController = TVApplicationController(context: appControllerContext, window: window, delegate: self)
        ...
        return true
    }
}
class AppDelegate: ..., TVApplicationControllerDelegate {
    ...
    static let tvBaseURL = "http://localhost:9001/"
    static let tvBootURL = "\(AppDelegate.tvBaseURL)/application.js"
    func application(..., didFinishLaunchingWithOptions launchOptions: ...) -> Bool {
        ...
        let appControllerContext = TVApplicationControllerContext()
        if let javaScriptURL = NSURL(string: AppDelegate.tvBootURL) {
            appControllerContext.jsApplicationURL = javaScriptURL
        }
        ...
        appController = TVApplicationController(context: appControllerContext, window: window,
            delegate: self)
        ...
        return true
    }
}
TVML and Styles
TVML and Styles

Markup to define template
TVML and Styles

Markup to define template
Templates have default style
TVML and Styles

Markup to define template
Templates have default style
Customize appearance with styles
<?xml version="1.0" encoding="UTF-8" ?>
<document>
  <head>
    <style>
      <!-- custom styles -->
    </style>
  </head>
  <templateName>
    <!-- template content -->
  </templateName>
</document>
<stackTemplate>
  <collectionList>
    <carousel>...</carousel>
    <shelf>...</shelf>
    ...
  </collectionList>
</stackTemplate>
// Configure text elements

<document>
    <head>
        <style>
            .lockupTitle {
                tv-text-highlight-style: marquee-and-show-on-highlight;
            }
        </style>
    </head>
    <stackTemplate>
        ...
        <lockup>
            <img src="..." width="..." height="..." />
            <title class="lockupTitle">...</title>
        </lockup>
        ...
    </stackTemplate>
</document>
The Transformers Collection

Transformers: Revenge of the Fallen

Price: $39.99
Buy
Wish List
<table>
<thead>
<tr>
<th>Title</th>
<th>Subtitle</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title 1</td>
<td>Subtitle 1</td>
<td>Text</td>
</tr>
<tr>
<td>Title 2</td>
<td>Subtitle 2</td>
<td>Text</td>
</tr>
<tr>
<td>Title 3</td>
<td>Subtitle 3</td>
<td>Text</td>
</tr>
<tr>
<td>Title 4</td>
<td>Subtitle 4</td>
<td>Text</td>
</tr>
<tr>
<td>Title 5</td>
<td>Subtitle 5</td>
<td>Text</td>
</tr>
<tr>
<td>Title 6</td>
<td>Subtitle 6</td>
<td>Text</td>
</tr>
</tbody>
</table>

Lorem ipsum dolor sit amet, ligula suspendisse nulla pretium, rhoncus tempor placerat fermentum, enim integer ad... MORE
TVMLKit JS
TVMLKit JS

XMLHttpRequest
DOM
Storage
TVMLKit JS

<table>
<thead>
<tr>
<th>XMLHttpRequest</th>
<th>Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOM</td>
<td>Device</td>
</tr>
<tr>
<td>Storage</td>
<td>Restrictions</td>
</tr>
<tr>
<td>App</td>
<td>Keyboard</td>
</tr>
<tr>
<td>NavigationDocument</td>
<td></td>
</tr>
<tr>
<td>MenuBarDocument</td>
<td></td>
</tr>
<tr>
<td>Slideshow</td>
<td></td>
</tr>
<tr>
<td>TVMLKit JS</td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td></td>
</tr>
<tr>
<td>XMLHttpRequest</td>
<td></td>
</tr>
<tr>
<td>Settings</td>
<td></td>
</tr>
<tr>
<td>DOM</td>
<td></td>
</tr>
<tr>
<td>Device</td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td></td>
</tr>
<tr>
<td>Restrictions</td>
<td></td>
</tr>
<tr>
<td>App</td>
<td></td>
</tr>
<tr>
<td>Keyboard</td>
<td></td>
</tr>
<tr>
<td>NavigationDocument</td>
<td></td>
</tr>
<tr>
<td>MediaItem</td>
<td></td>
</tr>
<tr>
<td>MenuBarDocument</td>
<td></td>
</tr>
<tr>
<td>Playlist</td>
<td></td>
</tr>
<tr>
<td>Slideshow</td>
<td></td>
</tr>
<tr>
<td>Player</td>
<td></td>
</tr>
</tbody>
</table>
TVMLKit JS
App lifecycle

When TVApplicationController is initialized

```javascript
App.onLaunch = function(options) {
    var locationStr = options['location'];
    ...
};
```
App error handler

Capture uncaught exceptions

```javascript
App.onError = function(message, sourceURL, line) {
    console.error("Uncaught Exception!", message, sourceURL, line);
};
```
TVMLKit JS

NavigationDocument

Manage document hierarchy

Global instance

Dismiss handled by framework
TVMLKit JS

NavigationDocument

Manage document hierarchy
Global instance
Dismiss handled by framework

```javascript
navigationDocument.pushDocument(document)
navigationDocument.replaceDocument(document, oldDocument)
navigationDocument.presentModal(document)
```
let loadingMarkup = `<?xml version="1.0" encoding="UTF-8" ?>
<document>
  <loadingTemplate>
    <activityIndicator>
      <text>Loading ...</text>
    </activityIndicator>
  </loadingTemplate>
</document>`;
let loadingDocument = new DOMParser().parseFromString(loadingMarkup, 'application/xml');
navigationDocument.pushDocument(loadingDocument);

// Later ...
let stackDocument = createStackDocument(...);
navigationDocument.replaceDocument(stackDocument, loadingDocument);
// Presenting documents

let loadingMarkup = `<?xml version="1.0" encoding="UTF-8" ?>
  <document>
    <loadingTemplate>
      <activityIndicator>
        <text>Loading ...</text>
      </activityIndicator>
    </loadingTemplate>
  </document>``;

let loadingDocument = new DOMParser().parseFromString(loadingMarkup, 'application/xml');

``navigationDocument.pushDocument(loadingDocument);

// Later ...

let stackDocument = createStackDocument(...);

navigationDocument.replaceDocument(stackDocument, loadingDocument);`
let loadingMarkup = `<?xml version="1.0" encoding="UTF-8" ?>
<document>
  <loadingTemplate>
    <activityIndicator>
      <text>Loading ...</text>
    </activityIndicator>
  </loadingTemplate>
</document>`;
let loadingDocument = new DOMParser().parseFromString(loadingMarkup, 'application/xml');
navigationDocument.pushDocument/loadingDocument);

// Later ...
let stackDocument = createStackDocument(...);
navigationDocument.replaceDocument(stackDocument, loadingDocument);
TVMLKit JS

NavigationDocument

Manage document hierarchy
Global instance
Dismiss handled by framework

```
navigationDocument.pushDocument(document)
navigationDocument.replaceDocument(document, oldDocument)
navigationDocument.presentModal(document)
```
Media Playback
// Setting up a TVMLKit JS Video Player

var video = new MediaItem('video', 'https://example.com/video.m3u8');
video.title = 'My Great Movie';
video.description = 'An extensive description...';
video.resumeTime = 10.0; // seconds

var playlist = new Playlist();
playlist.push(video);

var player = new Player();
player.playlist = playlist;
player.play(); // Present the player
// Setting up a TVMLKit JS Video Player

```javascript
var video = new MediaItem('video', 'https://example.com/video.m3u8');
video.title = 'My Great Movie';
video.description = 'An extensive description...';
video.resumeTime = 10.0; // seconds

var playlist = new Playlist();
playlist.push(video);

var player = new Player();
player.playlist = playlist;
player.play(); // Present the player
```
// Setting up a TVMLKit JS Video Player

var video = new MediaItem('video', 'https://example.com/video.m3u8');
video.title = 'My Great Movie';
video.description = 'An extensive description...';
video.resumeTime = 10.0; // seconds

var playlist = new Playlist();
playlist.push(video);

var player = new Player();
player.playlist = playlist;
player.play(); // Present the player
// Setting up a TVMLKit JS Video Player

var video = new MediaItem('video', 'https://example.com/video.m3u8');
video.title = 'My Great Movie';
video.description = 'An extensive description...';
video.resumeTime = 10.0; // seconds

var playlist = new Playlist();
playlist.push(video);

var player = new Player();
player.playlist = playlist;
player.play(); // Present the player
Zootopia
1 min
Disney presents a heartwarming comedy-adventure set in the modern mammal metropolis of Zootopia. With habitat neighborhoods like ritzy Sahara Square and frigid Tundratown, it’s a melting pot where animals from every environment live together—a place where no matter what you are, from the biggest elephant to the smallest shrew, you can be anything. But when optimistic Officer Judy Hopps arrives, she discovers that being the first bunny on a police force of big, tough... MORE
// Setting up a TVML JS Audio Player

var audio = new MediaItem('audio', 'https://example.com/audio.mp3');
audio.title = 'My Great Song';
audio.artworkImageURL = 'https://example.com/audio-artwork.jpg';

var playlist = new Playlist();
playlist.push(audio);

var player = new Player();
player.playlist = playlist;
player.play(); // Present the player
func application(application: UIApplication, didFinishLaunchingWithOptions launchOptions: [NSObject: AnyObject]?) -> Bool {
    // ...
    let audioSession = AVAudioSession.sharedInstance()
    let success = try? audioSession.setCategory(AVAudioSessionCategoryPlayback)
    // ...
}
Demo
Building TVMLKit application

Jeff Tan-Ang tvOS Design Engineer
Recap

Setup project using Xcode
Recap

Setup project using Xcode
Construct and manage documents
Recap

Setup project using Xcode
Construct and manage documents
Handle events
Recap

Setup project using Xcode
Construct and manage documents
Handle events
Playback videos
New Features

Parry Panesar tvOS Engineer
New Styles and Attributes

Animatable DOM Updates

ECMAScript 6

Light and Dark Appearance

Custom Collection Cells

Embedded Video

Image Placeholders

Audio Now Playing

Slideshow

New Player API’s

Interactive Video Overlays

New Lockups

Multi-row Shelf

Web Inspector Enhancements

TVMLKit
TVMLKit

- New Styles and Attributes
- Embedded Video
- New Lockups
- Animatable DOM Updates
- Image Placeholders
- Audio Now Playing
- ECMAScript 6
- Multi-row Shelf
- Light and Dark Appearance
- Slideshow
- Interactive Video Overlays
- Custom Collection Cells
- New Player API’s
- Web Inspector Enhancements
New Styles and Attributes
Embedded Video
New Lockups

Animatable DOM Updates
Image Placeholders
Multi-row Shelf

ECMAScript 6

Light and Dark Appearance
Slideshow
Interactive Video Overlays

Custom Collection Cells
New Player API’s
Web Inspector Enhancements

TVMLKit
Waterfall Relaxation
TVMLKit

- New Styles and Attributes
- Embedded Video
- New Lockups
- Animatable DOM Updates
- Image Placeholders
- Audio Now Playing
- ECMAScript 6
- Multi-row Shelf
- Light and Dark Appearance
- Slideshow
- Interactive Video Overlays
- Custom Collection Cells
- New Player API’s
- Web Inspector Enhancements
Light / Dark

Top Movies
- LONDON HAS FALLEN
- 10 CLOVERFIELD LANE
- BURNT
- DEADPOOL
- HOW TO BE SINGLE
- 13 HOURS

New & Noteworthy
- MOTHERLESS BROOKLYN
- SALLY FIELDS
- EDDIE THE EAGLE
- KUNG FU PANDA
- AWAY DAYS

$9.99 in HD—Limited Time
// tv-theme media query property

<document>
  <head>
    <style>
      .foo { color: rgb(0, 0, 0); }
    </style>
  </head>
  <templateName>
    ...
    <title class="foo">text</title>
    ...
  </templateName>
</document>
// tv-theme media query feature

<document>
  <head>
    <style>
      @media tv-template and (tv-theme:light) {
        .foo { color: rgb(0, 0, 0); }
      }
    }
    @media tv-template and (tv-theme:dark) {
      .foo { color: rgb(255, 255, 255); }
    }
  </style>
  </head>
  <templateName>
    ...
    <title class="foo">text</title>
    ...
  </templateName>
</document>
Embedded Video Playback
print("Hello, WWDC!")
learnAbout(OSX, iOS, watchOS, tvOS)
watchNow()
print("Hello, WWDC!")
learnAbout(OSX, iOS, watchOS, tvOS)
watchNow()
Embedded Video Playback

Playback context inside templates
Embedded Video Playback

Playback context inside templates
Plays on focus, or always
Embedded Video Playback

Playback context inside templates
Plays on focus, or always
Transitions to full screen
Embedded Video Playback

Playback context inside templates
Plays on focus, or always
Transitions to full screen
Supports all player functionality
Embedded Video Playback

Steps to configure

Configure template
Embedded Video Playback

Steps to configure

Configure template
Configure player
Embedded Video Playback

Steps to configure

Configure template
Configure player
Transition to fullscreen
Embedded Video Playback
Configure template

Wrap `<img>` with `<mediaContent>`
Embeded Video Playback
Configure template

Wrap `<img>` with `<mediaContent>`

Configure playback mode
Embedded Video Playback
Configure template

Wrap `<img>` with `<mediaContent>`

Configure playback mode

```
<lockup>
  <mediaContent playbackMode="onFocus">
    <img src="http://host/image.jpg" width="548" height="308"/>
  </mediaContent>
</lockup>
```
Embedded Video Playback
Configure template

Wrap `<img>` with `<mediaContent>`

Configure playback mode

```html
<lockup>
  <mediaContent playbackMode="onFocus">
    <img src="http://host/image.jpg" width="548" height="308"/>
  </mediaContent>
</lockup>
```
Embedded Video Playback

Configure template

Wrap `<img>` with `<mediaContent>`

Configure playback mode

```html
<lockup>
  <mediaContent playbackMode="onFocus">
    <img src="http://host/image.jpg" width="548" height="308"/>
  </mediaContent>
</lockup>
```
Embedded Video Playback
Configure player

Player per <mediaContent>

Access with ‘Player’ feature
Embedded Video Playback

Configure player

Player per `<mediaContent>`

Access with ‘Player’ feature

```javascript
var mediaContentElement = document.getElementsByTagName('mediaContent').item(0);
var player = mediaContentElement.getFeature('Player');
player.playlist = playlist;
```
Embedded Video Playback

Transition to fullscreen

Does not transition automatically
Embedded Video Playback

Transition to fullscreen

Does not transition automatically

document.addEventListener('select', function(event) {
    var mediaContentElement = event.target.getElementsByTagName('mediaContent').item(0);
    var player = mediaContentElement.getFeature('Player');
    player.present();
});
Embedded Video Playback

Transition to fullscreen

Does not transition automatically

document.addEventListener('select', function(event) {
    var mediaContentElement = event.target.getElementsByTagName('mediaContent').item(0);
    var player = mediaContentElement.getFeature('Player');
    player.present();
});
Embedded Video Playback

Transition to fullscreen

Does not transition automatically

```javascript
document.addEventListener('select', function(event) {
    var mediaContentElement = event.target.getElementsByTagName('mediaContent').item(0);
    var player = mediaContentElement.getFeature('Player');
    player.present();
});
```
Interactive Video Overlays
Buy Day Pass
Starting at $169

Buy 4 Day Pass
Starting at $399

Buy 7 Day Pass
Starting at $609
Interactive Video Overlays

One per player
Use any template
Presented on full screen video
Interactive Video Overlays

One per player
Use any template
Presented on full screen video

// Set a document to present over the player.
player.modalOverlayDocument = document;
Demo
Appearance, embedded video and interactive video overlay

Jeff Tan-Ang tvOS Design Engineer
Recap

Adopt dark appearance
Recap

Adopt dark appearance

Immersive using embedded videos
Recap

- Adopt dark appearance
- Immersive using embedded videos
- Interactive video overlays
Summary

Native experience
Summary

Native experience
Rapid development
Summary

Native experience
Rapid development
New features
More Information

https://developer.apple.com/wwdc16/212
## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Venue</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Designing for tvOS</td>
<td>Presidio</td>
<td>Tuesday 4:00PM</td>
</tr>
<tr>
<td>Mastering UIKit on tvOS</td>
<td>Presidio</td>
<td>Wednesday 10:00AM</td>
</tr>
<tr>
<td>Focus Interaction on tvOS</td>
<td>Mission</td>
<td>Wednesday 4:00PM</td>
</tr>
<tr>
<td>Developing tvOS Apps Using TVMLKit: Part 2</td>
<td>Mission</td>
<td>Thursday 4:00PM</td>
</tr>
<tr>
<td>Labs</td>
<td>Frameworks Lab D</td>
<td>Wednesday 2:00PM</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>tvOS Lab</td>
<td>Frameworks Lab D</td>
<td>Wednesday 2:00PM</td>
</tr>
<tr>
<td>TVMLKit Lab</td>
<td>Graphics, Games, and Media Lab B</td>
<td>Wednesday 3:00PM</td>
</tr>
<tr>
<td>tvOS Lab</td>
<td>Frameworks Lab D</td>
<td>Thursday 9:00AM</td>
</tr>
<tr>
<td>TVMLKit Lab</td>
<td>Graphics, Games, and Media Lab C</td>
<td>Friday 9:00AM</td>
</tr>
</tbody>
</table>