

# MPMusicPlayerController & MPMediaPickerController

Fabrice.Kordon@lip6.fr





# As an introduction...

## **MPMusicPlayerController**

- Management of play lists
  - ▶ Play
  - ▶ Navigate,
  - ▶ etc.
- Queue of pieces
  - ▶ From the device's library (predefined queries)
  - ▶ From a dedicated playlist (MPMediaItemCollection)



## **MPMediaPickerController**

- Access to the device's music library
  - ▶ Selection of dedicated pieces
- Similar to...
  - ▶ Camera, photo album, address book...





# MPMusicPlayerController — preparation

## Activation

- Retrieve the common player from a class variable
  - ▶ `applicationMusicPlayer`
  - ▶ `systemMusicPlayer` (Apple's Music App, receives remote control events)

## Installing a playlist

- Building the list from the device's music library

```
func setQueue(with query: MPMediaQuery)
```

- ▶ query of type `MPMediaQuery`

```
init(filterPredicates: Set<MPMediaPredicate>?)
class func albums() -> MPMediaQuery
class func artists() -> MPMediaQuery
class func songs() -> MPMediaQuery
class func playlists() -> MPMediaQuery
class func podcasts() -> MPMediaQuery
class func audiobooks() -> MPMediaQuery
class func compilations() -> MPMediaQuery
class func composers() -> MPMediaQuery
class func genres() -> MPMediaQuery
```



- Building from existing `MPMediaItemCollection`

```
func setQueue(with itemCollection: MPMediaItemCollection)
```



# MPMusicPlayerController — management

4



## Navigation

```
func skipToNextItem()  
func skipToPreviousItem()  
func skipToBeginning()
```



## Modifying the playlist

- After the last media / after the currently playing item

```
func append(_ descriptor: MPMusicPlayerQueueDescriptor)  
func prepend(_ descriptor: MPMusicPlayerQueueDescriptor)
```



## Playback state & mode

- Properties
  - ▶ `nowPlayingItem`, `indexOfNowPlayingItem`
  - ▶ `repeatMode`, `playbackState`, `shuffleMode`



# MPMusicPlayerController — management

4



## Navigation

```
func skipToNextItem()  
func skipToPreviousItem()  
func skipToBeginning()
```



## Modifying the playlist

After the last method, you can also use the following methods:

```
func append(_ desc: String)  
func prepend(_ desc: String)
```



**Which thread?**

On the Apps main thread only!



## Playback state & mode

Properties

- ▶ `nowPlayingItem`, `indexOfNowPlayingItem`
- ▶ `repeatMode`, `playbackState`, `shuffleMode`



# MPMusicPlayerController — playing

5

## Protocol MPMediaPlayback

- Available in MPMusicPlayerController
- Numerous functions
  - ▶ `play()`
  - ▶ `pause()`
  - ▶ `stop()`
  - ▶ `prepareToPlay()`
  - ▶ `beginSeekingBackward()`
  - ▶ `beginSeekingForward()`
  - ▶ `endSeeking()`
- Some properties
  - ▶ `currentPlaybackRate`
  - ▶ `currentPlaybackTime`



# MPMusicPlayerController — notifications

6



## Notification system to control playback

- Activate / deactivate
  - ▶ `beginGeneratingPlaybackNotifications`
  - ▶ `endGeneratingPlaybackNotifications`
- Available notifications
  - ▶ `MPMusicPlayerControllerPlaybackStateDidChange`
  - ▶ `MPMusicPlayerControllerNowPlayingItemDidChange`
  - ▶ `MPMusicPlayerControllerVolumeDidChange`

## You already know how to get such notifications

### ▶ Via the notification center

```
func addObserver(forName name: NSNotification.Name?,
                 object obj: Any?,
                 queue: OperationQueue?,
                 using block: @escaping (Notification) -> Void) -> NSObjectProtocol
func addObserver(_ observer: Any,
                 selector aSelector: Selector,
                 name aName: NSNotification.Name?,
                 object anObject: Any?)
```



# MPMediaPickerController

7

## Behaves like other «pickers»

- Creation
  - ▶ À la Swift ou à la Objective-C
- Presented like other pickers
- Customisation by means of properties
  - ▶ `allowsPickingMultipleItems`
  - ▶ `showsCloudItems`
  - ▶ `mediaTypes`
  - ▶ `prompt`
  - ▶ `showsItemsWithProtectedAssets`

## Feed-back handled by `MPMediaPickerControllerDelegate`

- Do not forget to set the delegate



# MPMediaPickerControllerDelegate

8

## Two optional methods

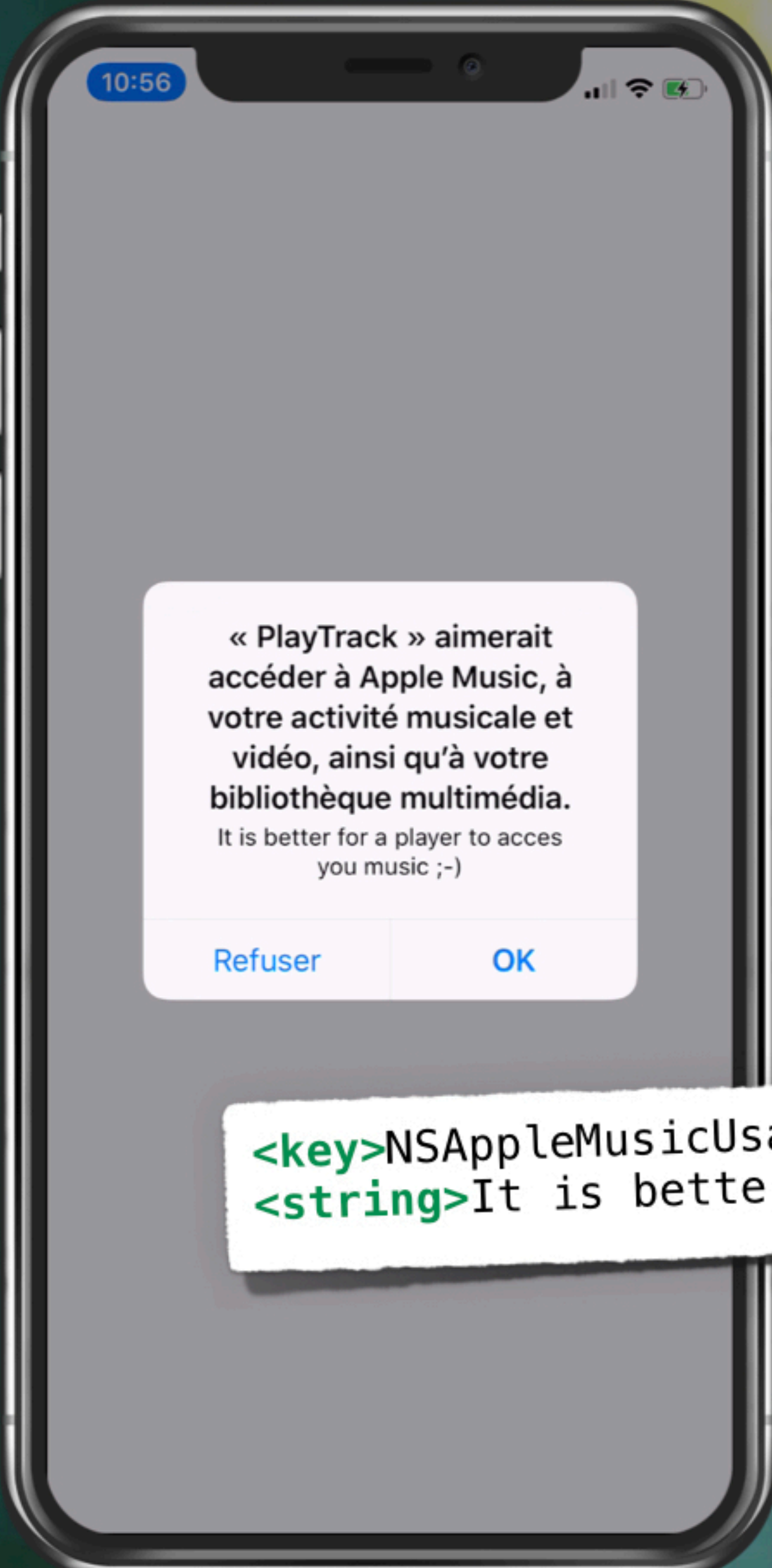
```
func mediaPicker(_ mediaPicker: MPMediaPickerController,  
                didPickMediaItems mediaItemCollection: MPMediaItemCollection)  
  
func mediaPickerDidCancel(_ mediaPicker: MPMediaPickerController)
```

• Trivial to use

▶ You already saw this



# Privacy



## Since iOS10

- You must declare the access to the music library
- info.plist



## Otherwise?

- It does not work

```
<key>NSAppleMusicUsageDescription</key>  
<string>It is better for a player to access you music ;-)</string>
```



# As a conclusion...



## Easy is'n't it?

- `MPMediaPickerController`
  - ▶ Browse and select music in the device's music library
  - ▶ Build playlist
  - ▶ All included (cloud access, media type selection, etc.)
- `MPMusicPlayerController`
  - ▶ Navigate through a play list
  - ▶ Play music



## Remember to import the framework

- `MediaPlayer`

