

A bit about CarPlay

Fabrice.Kordon@lip6.fr

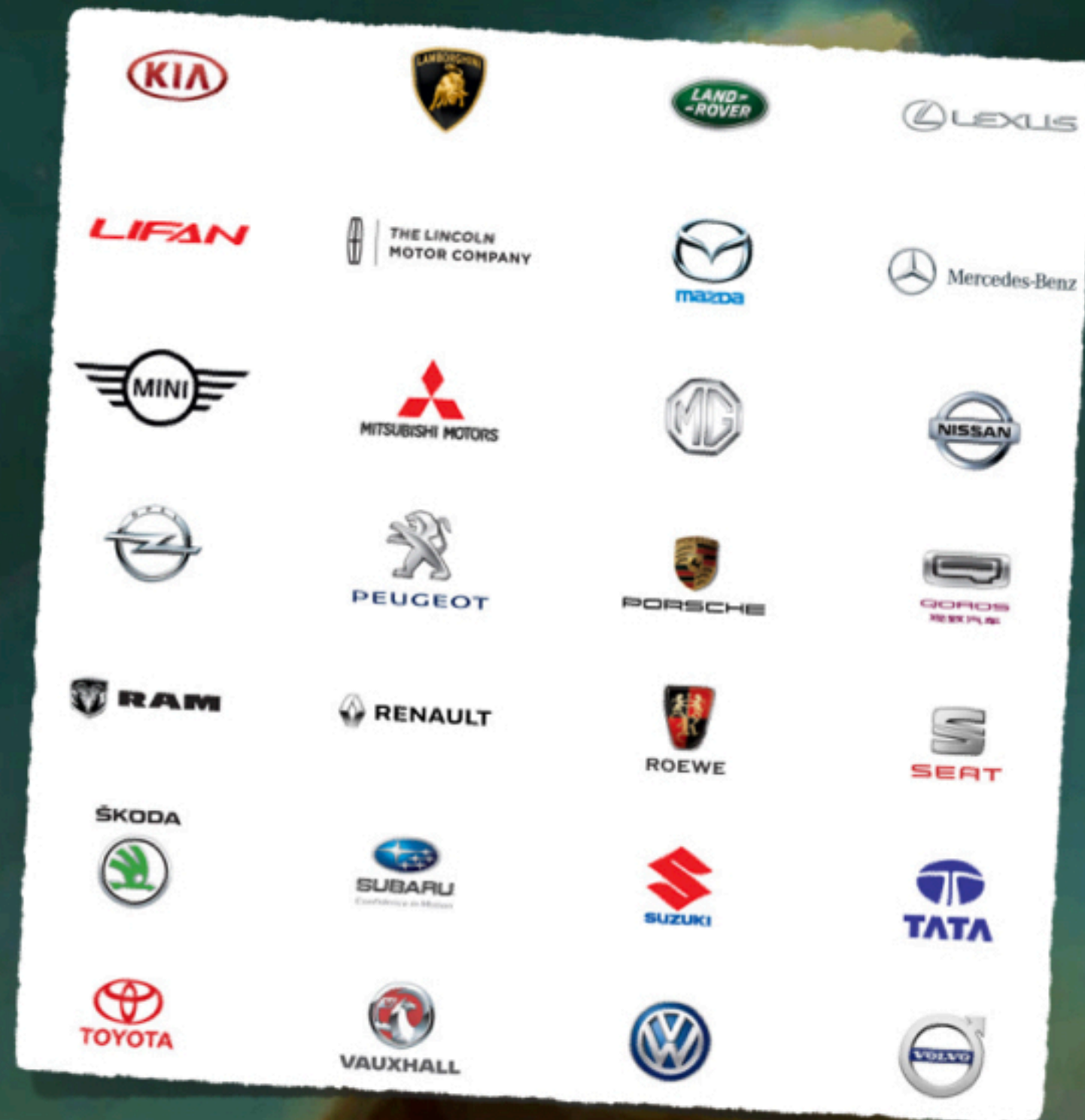


As an introduction...



Embedded version of iOS for cars

- First presented @WWDC'2013
 - ▶ But really working on iOS9 (September 2015)
 - ▶ Now commonly available



As an introduction...

Embedded version of iOS for cars

- First present
- ▶ But really w
- ▶ Now commo



Objective!

Give you a rough idea of CarPlay



Most of them also support Android Auto



Business is business!



Features

Use of a secondary screen

- Provided by the Car maker
 - ▶ Introduces numerous sizes
- Simplified interface
 - ▶ Secure driving

Deported controls

- Those of the car
- Possibly touch screen (if available)

Three types of Apps

- Spoken Audio apps
 - ▶ Music player, telephone, voice-controlled apps
- Navigation apps

Features

Use of a secondary screen

- Provided by the Car maker
 - ▶ Introduces numerous features
- Simplified interface
 - ▶ Secure driving



In which framework?

CarPlay of course!

Deported content

- Those of the car
- Possibly touch screen (if available)

Three types of Apps

- Spoken Audio apps
 - ▶ Music player, telephone, voice-controlled apps
- Navigation apps

Constraints

AppDelegate answers to CPApplicationDelegate

- A Protocol
- Two mandatory methods

```
func application(_ application: UIApplication,  
                didConnectCarInterfaceController  
                interfaceController: CPInterfaceController,  
                to window: CPWindow)
```

```
func application(_ application: UIApplication,  
                didDisconnectCarInterfaceController  
                interfaceController: CPInterfaceController,  
                from window: CPWindow)
```

- ▶ A dedicated interface controller provided
- ▶ A dedicated window provided
- ▶ Suggestion, keep it safe in variables

- A few more methods



Constraints

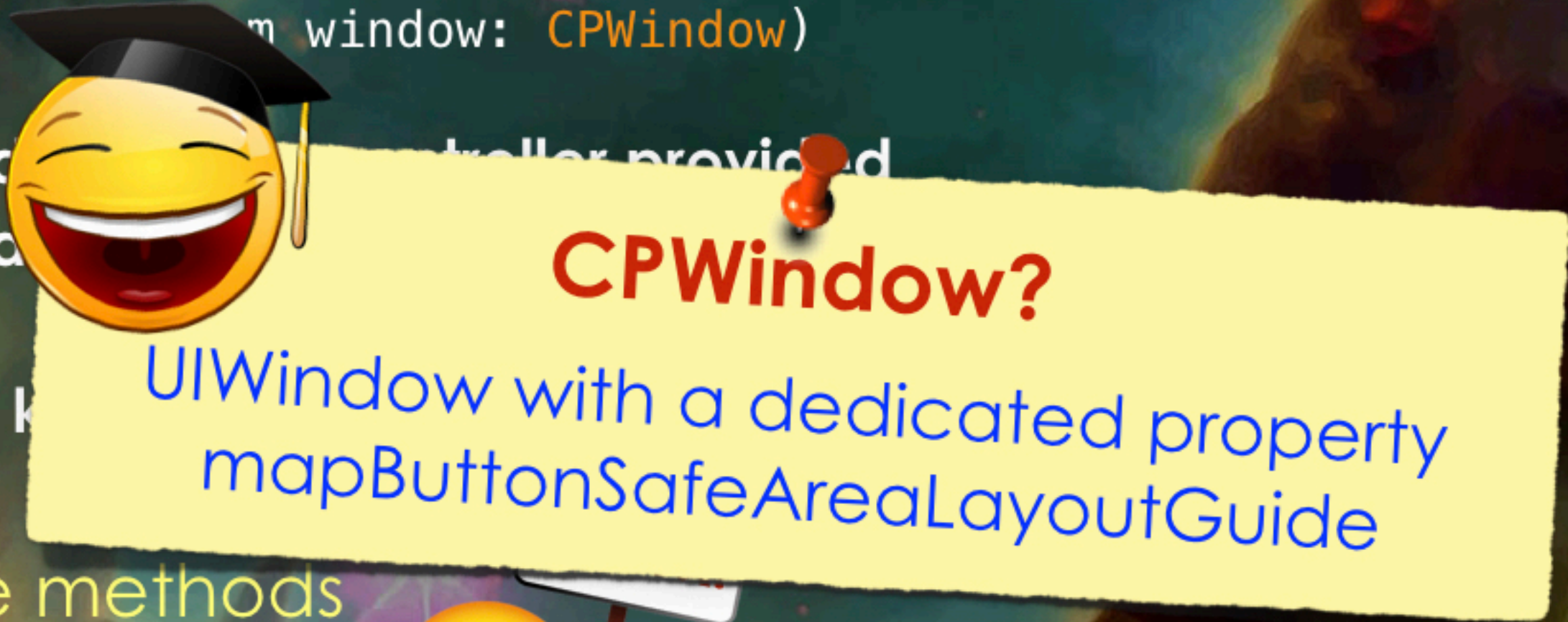
AppDelegate answers to CPApplicationDelegate

- A Protocol
- Two mandatory methods

```
func application(_ application: UIApplication,  
                didConnectCarInterfaceController  
                interfaceController: CPInterfaceController,  
                to window: CPWindow)
```

```
func application(_ application: UIApplication,  
                didDisconnectCarInterfaceController  
                interfaceController: CPInterfaceController,  
                from window: CPWindow)
```

- ▶ A dedicated controller provided
- ▶ A dedicated
- ▶ Suggestion, k



CPWindow?
UIWindow with a dedicated property
mapButtonSafeAreaLayoutGuide

- A few more methods



CPInterfaceController

📱 Handles the deported interface

👤 All is managed by means of templates

▶ Properties

```
var rootTemplate: CPTemplate { get }
var topTemplate: CPTemplate? { get }
var templates: [CPTemplate] { get }
var presentedTemplate: CPTemplate? { get }
```

▶ Template manipulation

```
func pushViewController(_ templateToPush: CPTemplate,
                    animated: Bool)
func popTemplate(animated: Bool)
func popToRootTemplate(animated: Bool)
func pop(to targetTemplate: CPTemplate,
        animated: Bool)
func dismissTemplate(animated: Bool)
```

📱 CPInterfaceControllerDelegate

👤 Dedicated actions

- ▶ When templates appears
- ▶ When template disappear

Navigation apps

6



CPMapTemplate

- Manages Map buttons
 - ▶ **Array of CPMapButton**
- Navigation sessions
- Alert display
- Panning of the map
- Route management



CPMapTemplateDelegate protocol

- Classes of handled events
 - ▶ **Navigation events (trip and route, navigation cancel, etc.)**
 - ▶ **Notifications**
 - ▶ **Navigation alerts**
 - ▶ **Map panning**
 - ▶ **A bit of gestures**

Spoken Audio apps

7

Connection with the MediaPlayer framework

MPPlayableContentManager

-  Management of playable content
-  Updating data
-  Handling currently playing items

MPPlayableContentDelegate


-  Handling beginning of play
-  Handling update of the playable content manager's context

As a conclusion...


Native device apps

-  UIKit & co
 - ▶ Display and execution handled on the device

Watch apps

-  WatchKit & co
 - ▶ Display and execution handled on the watch
 - ▶ Possibly some parts of the extension on the device too

CarPlay app

-  CarPlay & co
 - ▶ Display on device + deported (low interaction on device)
 - ▶ All execution on device
 - ▶ Support of remote control (from the car actuators)
 - ▶ Parameterised interface configuration
 - ▶ Limited interactivity