

UNNofification

Fabrice.Kordon@lip6.fr



As an introduction...



UNNotification?

- Implemented in the UserNotifications framework
- Communicate with the user
- Display messages...
- remind something
 - ▶ Agenda?
 - ▶ Regular action?

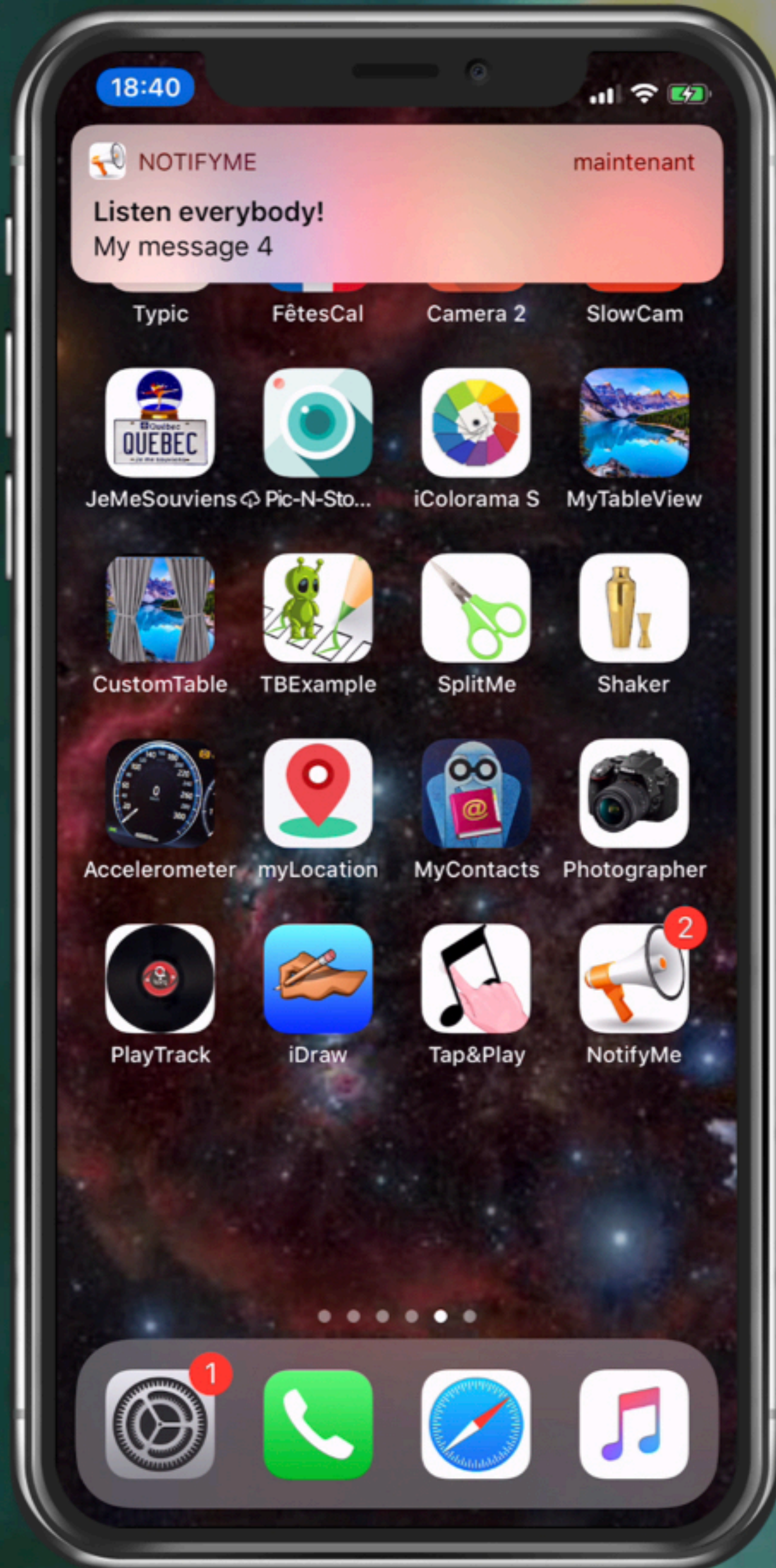


Visibility?

- According to the preferences

How does it work?

3



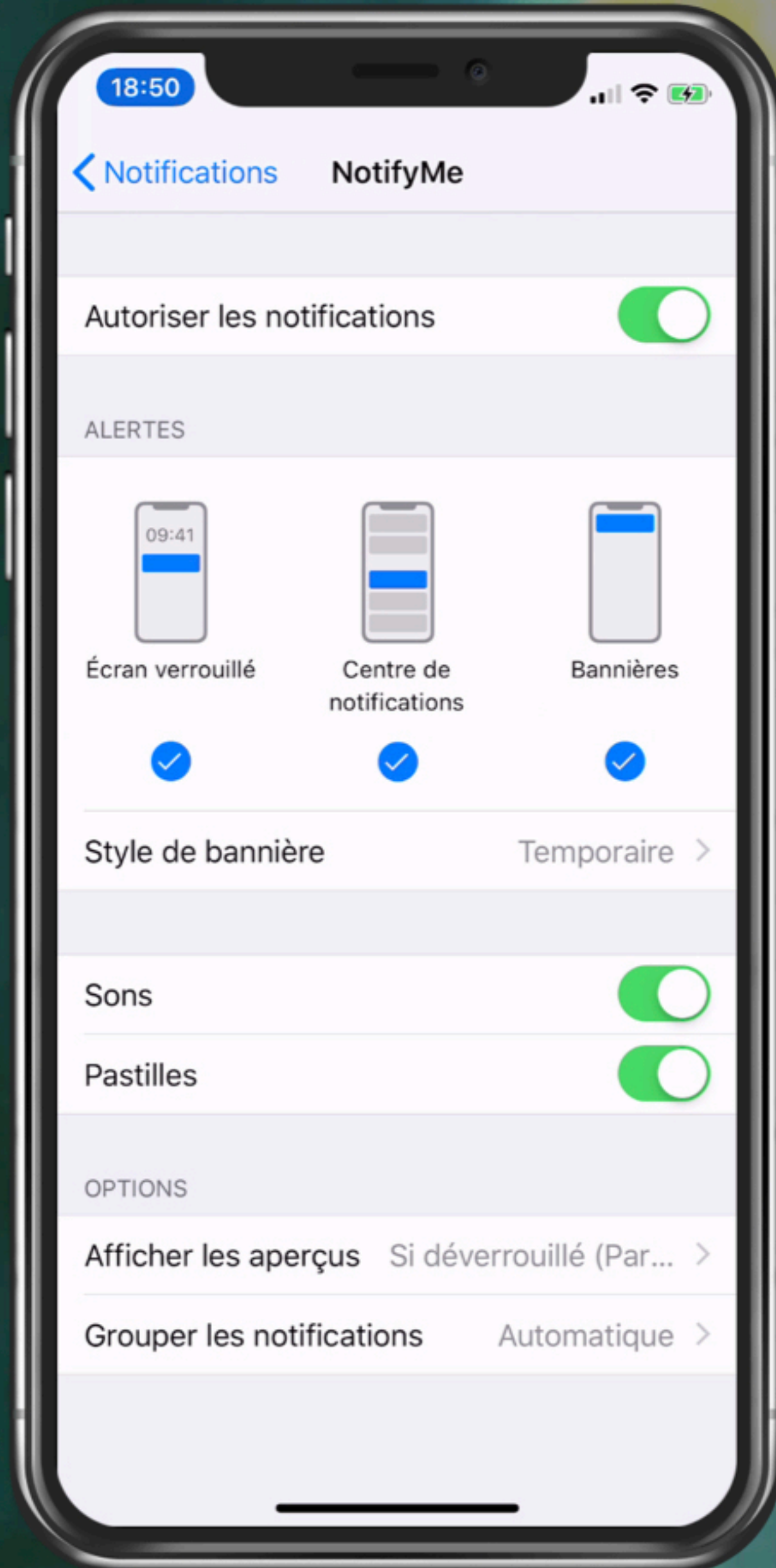
Your App sends a notification to the device

- Temporary display
 - ▶ Tou «tap» and it goes to your App
- Stored in the notification center
 - ▶ For later display
- When your App is launched
 - ▶ You may program a catch

Settings

- All in the preference panel
- Possible to configure App per App

How does it work?



Your App sends a notification to the device

- Temporary display
 - ▶ Tou «tap» and it goes to your App
- Stored in the notification center
 - ▶ For later display
- When your App is launched
 - ▶ You may program a catch

Settings

- All in the preference panel
- Possible to configure App per App

UNNotificationCenter

4

 **Allows to post UNNotifications**

 **Principles**

 Get a link to the shared UNNotificationCenter

▶ **Class method current()**

 Push notification requests

```
func add(_ request: UNNotificationRequest,  
        completionHandler: ((Error?) -> Void)? = nil)
```

 Get pending requests

```
func getPendingNotificationRequests(completionHandler: @escaping  
                                   ([UNNotificationRequest]) -> Void)
```

 Programmatic suppression possible too

```
func removePendingNotificationRequests(withIdentifiers identifiers: [String])
```

```
func removeAllPendingNotificationRequests()
```

UNNotificationRequest

Content

Three properties

- ▶ **identifier**, the unique identifier of this request
- ▶ **content** , its content
- ▶ **trigger**, the condition to deliver it (you do not manipulate it)

Creation

```
init(identifier: String,  
      content: UNNotificationContent,  
      trigger: UNNotificationTrigger?)
```

Handled by delegation

UNUserNotificationCenterDelegate

UNNotificationContent & UNMutableNotificationContent

6

Identical

- One can be modified

Content

- Properties title, subtitle, body

- ▶ Describe the content

- Properties sound

- ▶ Associated sound

- launchImageName

- ▶ Custom image to display

- userInfo

- ▶ Custom information (dictionary) associated to the notification

UNNotificationTrigger

📱 Usually used as subclasses

🕒 UNTimeIntervalNotificationTrigger

```
init(timeInterval: TimeInterval, repeats: Bool)
```

📅 UNCalendarNotificationTrigger

```
init(dateMatching dateComponents: DateComponent, repeats: Bool)
```

▶ For the two previous ones

```
func nextTriggerDate() -> Date?
```

📍 UNLocationNotificationTrigger

```
init(region: CLRegion, repeats: Bool)
```

📡 UNPushNotificationTrigger

▶ For Push notification (not discussed there)

📱 Usage?

🕒 creation and then insertion in the notification query

UNUserNotificationCenterDelegate

8

📱 Handling of the notification by the user

📱 Three methods

👤 Process a user's response to a delivered notification

```
func userNotificationCenter(_ center: UNUserNotificationCenter,
                           didReceive response: UNNotificationResponse,
                           completionHandler: @escaping () -> Void)
```

👤 Handling an action delivered while the App is in foreground

```
func userNotificationCenter(_ center: UNUserNotificationCenter,
                           willPresent notification: UNNotification,
                           completionHandler: @escaping
                           (UNNotificationPresentationOptions) -> Void)
```

▶ Presentation options : badge, sound, alert

👤 Display the preference panel for the App

```
func userNotificationCenter(_ center: UNUserNotificationCenter,
                           openSettingsFor notification: UNNotification?)
```

Privacy



You must request for an authorisation

```
func requestAuthorization(options: UNAuthorizationOptions = [],  
                          completionHandler: @escaping (Bool, Error?) -> Void)
```

UNAuthorizationOptions

- ▶ .badge
- ▶ .sound
- ▶ .alert
- ▶ .carPlay
- ▶ .criticalAlert

You may do it once or separately...



First time?



As a conclusion...



Nice and not so complex is'n't it?

- Some examples in future videos
 - ▶ Very simple
 - ▶ Simple (with some actions)



Apple Push Notification Service?

- Goes through a global notification center (by Apple)
 - ▶ This is strongly protected
- You need to state this in your developer certificate