SCIENCES SORBONNE UNIVERSITÉ

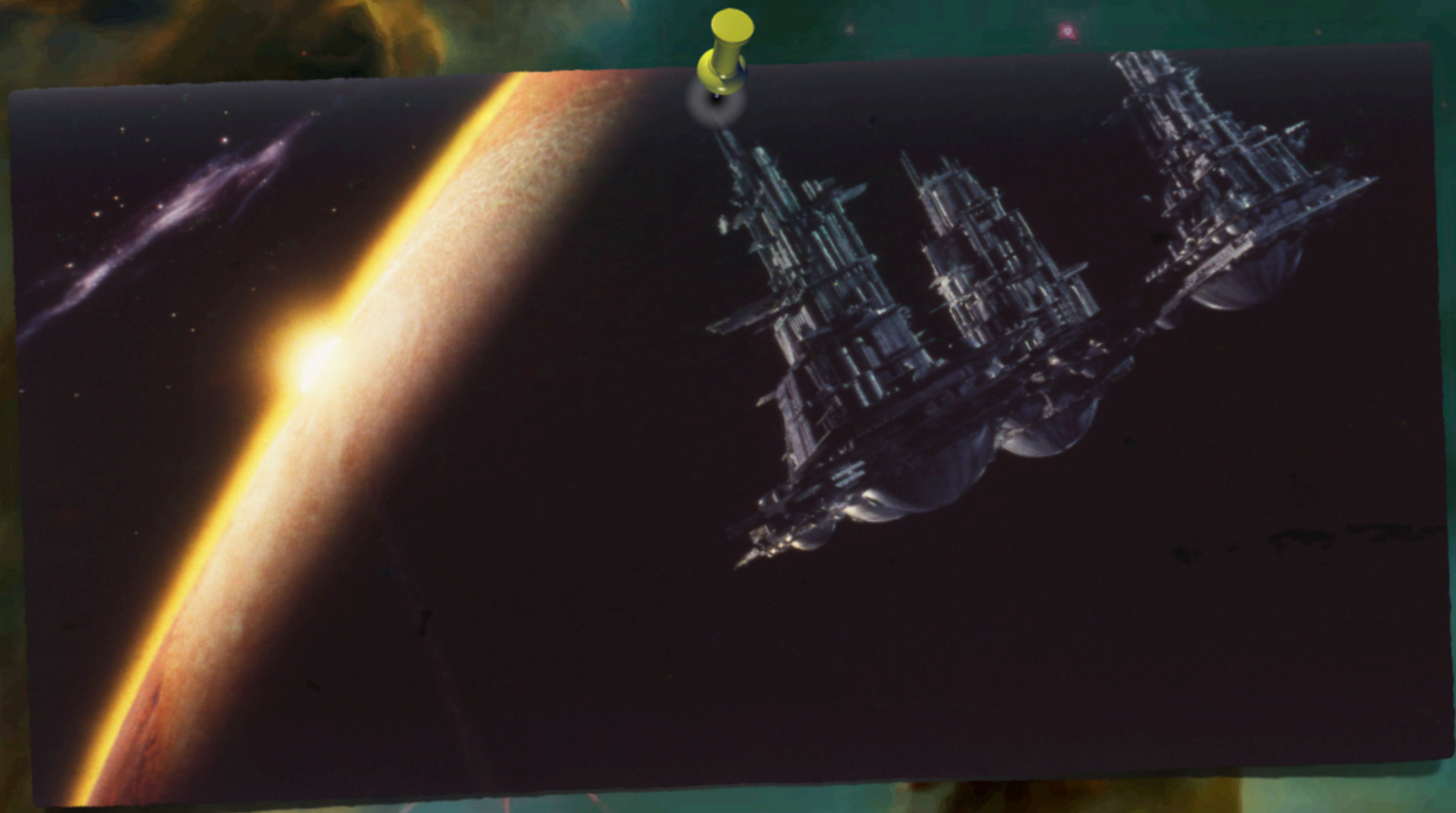
«Asteroids» in Swift

Fabrice.Kordon@lip6.fr

LIP6

As an introduction

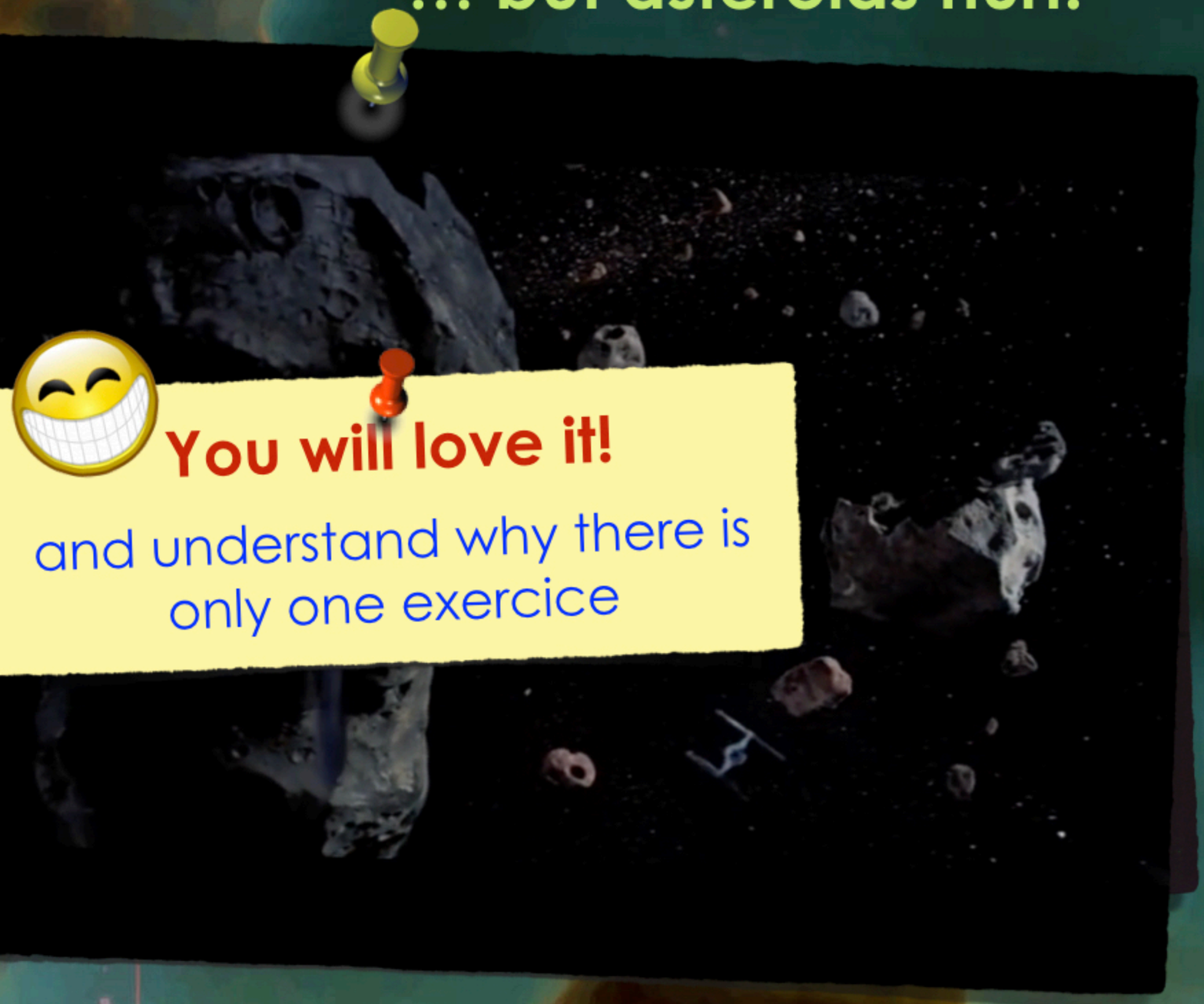
- In space, nobody can hear you scream...



As an introduction

In space, nobody can hear you scream...

... but asteroids hurt!



You will love it!

and understand why there is only one exercice

Demo (4” screen)

Demo (5.8” screen)

Demo (9.7” screen)

Some hints

A bit more difficult than previous exercice

- You reach one major milestone 

Let's detail a few tricks to help you

1 — multiviews in a monoview app

2 — several behavior for some views

3 — about GameView

4 — handling buttons to move the TIE

5 — Some strategic suggestions

1 — multiview in a monoview App

Stack is your friend

- ScoreView
  - Displays scores
  - Adds a best score
- PrefView
  - Choice of a difficulty
- GameView
  - Access to the game
  - Access to preferences
  - Access to score
  - Game
- ViewController
  - Main views + background & parallax effect
  - Controls other views

1 — multiview in a monoview App

Stack is your fri

- ScoreView
  - Displays scores
  - Adds a best score
- PrefView
  - Choice of a difficulty
- GameView
  - Access to the game
  - Access to preferences
  - Access to score
  - Game
- ViewController
  - Main views + backgr parallax effect
  - Controls other views

MVC!

Views request the Controller to be visible

hidden property

ScoreView & PrefView are hidden

Effects

Do not forget blur & parallax

2 — several behavior for some views

Some views supports several behaviors

- GameView
  - Entrance (button to play)
  - Access to preferences and scores
  - CAUTION : be careful when restarting a new play
- ScoreView
  - Differentiated behavior for small devices (room for the keyboard)
  - Be prepared to handle UITextFieldDelegate
  - Show current scores OR type a name for a new best 5 scores
  - Automatically activated when a game ends if necessary
- Do not forget isHidden
  - Even inside a view for dedicated elements

3 — about GameView

Having the game going on

- A timer of course
- Duration of the display when game ends
  - **A timer too**
- Some ideas for parameterization
  - **Max of asteroids depends on the device type and the level**
    - **basis = 20 for a small device, 50 for a large device**
    - **maxAsteroids = max + 10xlevel (numbered from 0 to 4)**
    - **probability of having a new asteroid = random % 100 + counter**
- Be careful with the event loop
  - **Compute new position for the TIE**
  - **Generate (if possible + random) a new asteroid**
  - **Move all existing asteroids + check for collision with TIE**
  - **Have asteroids out of the screen disappear (possibly recycled)**

4 — handling buttons to move the TIE

wo useful events

- .touchDown
  - when pressed to activate move
- .touchUpInside
  - when button released to deactivate move

Multiple association of targets

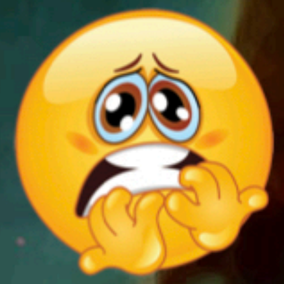
- Several targets for several events
  - start / stop moving
- Move handled in the event loop
  - With asteroids

5 — Some strategic suggestions

A bit of methodology…

How to develop classes

- ViewController
  - Create three empty views first with different background
  - Prepare the exchange logic between the three views
  - then…
- PrefView
  - The easy one
- ScoreView
  - A bit more tricky
- GameView
  - The «big one»
  - Do not forget a class to handle asteroids



As a conclusion…

- **There is some work to do…**
  - Prepare a bit of aspirin 😂
- **You will be pleased by the result**
  - And I want to spend practice session to play

- You may enrich (explosion, collisions with UIDynamics, etc.)
- Ressources in the companion web page of the video