

# REEEEEVIEW

Kenji Pa, 2014-07-02

# It's time to 找數

Projects I have worked on:

1. AFR
2. Piczle

Current working on:

- Retain.cc

# AFR: Lesson learnt

RTFM!

# Code Hunt

Consider the following code:

```
1.  some_id = SecureRandom.hex(5).upcase! if another_id.nil?  
2.  String.new(SOME_FORMAT) % {  
3.    [...]  
4.    some_id: some_id.ljust(10),  
5.  }
```

Line 4 throws a `NoMethodError: undefined method `ljust' for nil:NilClass` twice in 2 months.

Can you spot the problem?

# Cont: Code Hunt

Spent many time looking at this code and... surrender!

From Ruby's docs:

**upcase! → str or nil**

Upcases the contents of *str*, **returning nil** if no changes were made. Note: case replacement is effective only in ASCII region.

Sorry I should have read the document more carefully... :cry:

# Piczle

- First developed on Android, then move to iOS

If you have played Piczle, you know it crashes a lots on iOS.

- WRRRRRRRRRRRRRY!
- The first day when Roy QCed it on iOS was literally a DISASTER.

# Piczle: Crashes Explained

1. iOS has much stricter memory requirement than Android
  - a. around ~140mb on iPhone 5s, lower for older models
  - b. Memory requirement is even more stricter when you open up the camera... (> 120mb = crash)
  - c. Piczle UI images uses A LOTS memory
  - d. Memory leak leak

# Piczle: Memory leaked

- Many images is loaded dynamically and they are not managed by unity GC
  - As you guess I am not aware of it at the beginning...
- THE UI JUST USES A LOT OF MEMORY!
- The jigsaw module uses A LOT OF MEMORY (~30MB) and don't release itself. Good is that they do not increase so I can live with it.



# Piczle: Fixes&Optimizations

- Unload unused images and reduce cache used
  - It makes the UI more laggy, but better than crashes...
- Reduce size of UI image:
  - 2048\*4096 -> 2048\*2048, 1024\*1024 -> 512\*512
- Explicitly ask Unity & Mono to do GC before taking pictures

And these are not enough, since Unity does some internal caching and preload too...

- How do I know? List all resources used and review them one by one.  
(Resource.[FindObjectsOfTypeAll](#))
- Just deal with it! Explicitly unload those cache! (好孩子請勿模仿)



Q & A