

**THE FOLLOWING PRESENTATION MAY CONTAIN
EXCESSIVE USE OF ANIMATIONS
AND
IMAGES OF QUESTIONABLE TASTE
THOSE OF A SENSITIVE NATURE SHOULD LEAVE NOW**

What's New in iOS 7

Safari Services

Multipeer Connectivity

SpriteKit

UIKit Dynamics

AirDrop

TextKit

JavaScript Core

Multitasking

Modules

Game Controllers

What's New in iOS 7

Inter-App Audio OpenGL ES 3.0

NSURLComponents

iBeacons

NSURLSession

Message Attachments

Media Accessibility

NSProgress

What's New in iOS 7...

What's New in iOS 7 for Hipsters

What's New in iOS 7 for Hipsters



Until...



ios 7

Written by *Matth Thompson* on Sep 23rd, 2013

With the NDA finally lifted, we can finally talk about all of the amazing new APIs in iOS 7. And there are a *lot* of them. "1500 new APIs", by Apple's count during the WWDC Keynote. (Granted, a good portion of that could just be all of the changes from `id` to `instancetype`, but that's a huge number, regardless).

We'll be going over many of the new features iOS 7 in depth over the coming weeks, but with all of the excitement around this major release, this week's issue will hit on some of the gems hiding in plain sight: `NSData Base64 encoding`, `NSURLComponents`, `NSProgress`, `NSProcessInfo` activities, `CIDetectorSmile`, `CIDetectorEyeBlink`, `SSReadingList`, `AVCaptureMetaDataOutput`, `AVSpeechSynthesizer`, and `MKDistanceFormatter`.

What's New in iOS 7 for Hipsters



What's New in iOS 7 for the Hipster's Hipster

But first

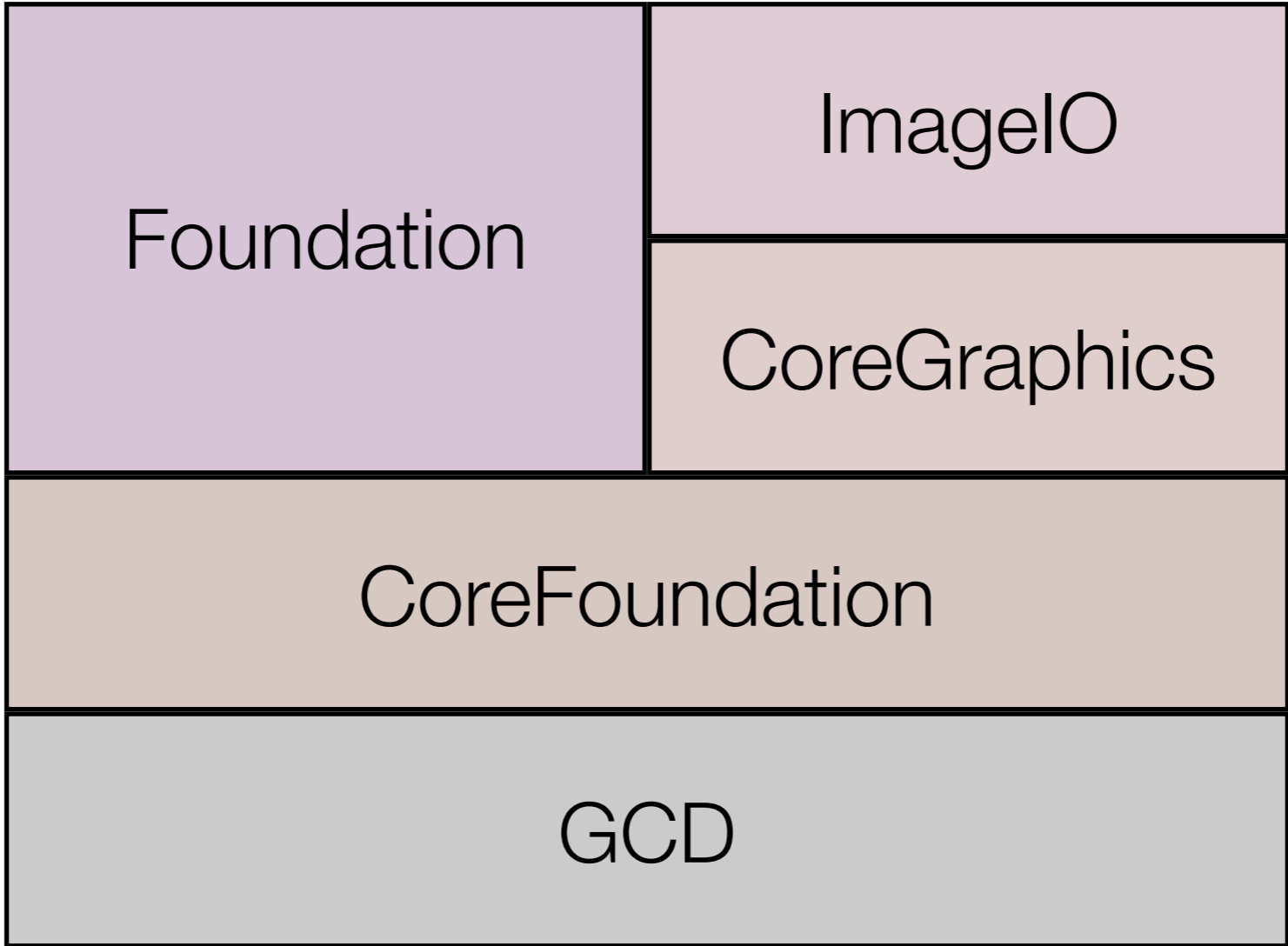
But first

A dirty secret

But first

A dirty secret

I'm a Mac developer



dispatch_data

dispatch_data

- Introduced in OS X 10.7 / iOS 5

dispatch_data

- Introduced in OS X 10.7 / iOS 5
- C-level immutable container for raw data, similar to NSData

Concatenating Data

NSData

NSData

Concatenating Data

NSMutableData

NSData

NSData

Concatenating Data



Concatenating Data



Concatenating Data



Concatenating Data



- Bytes are copied, potentially wastefully

Concatenating Data

dispatch_data_t

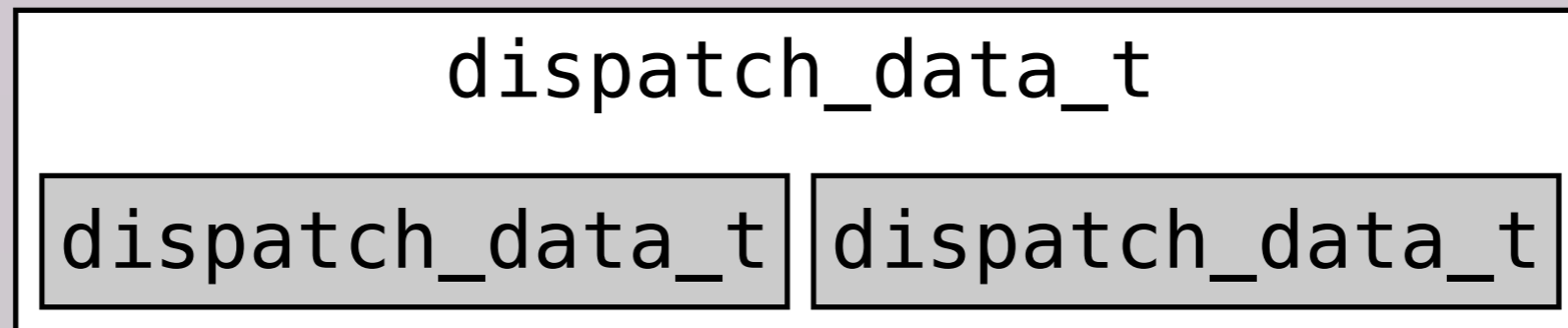
dispatch_data_t

Concatenating Data

```
dispatch_data_create_concat( dispatch_data_t dispatch_data_t )
```

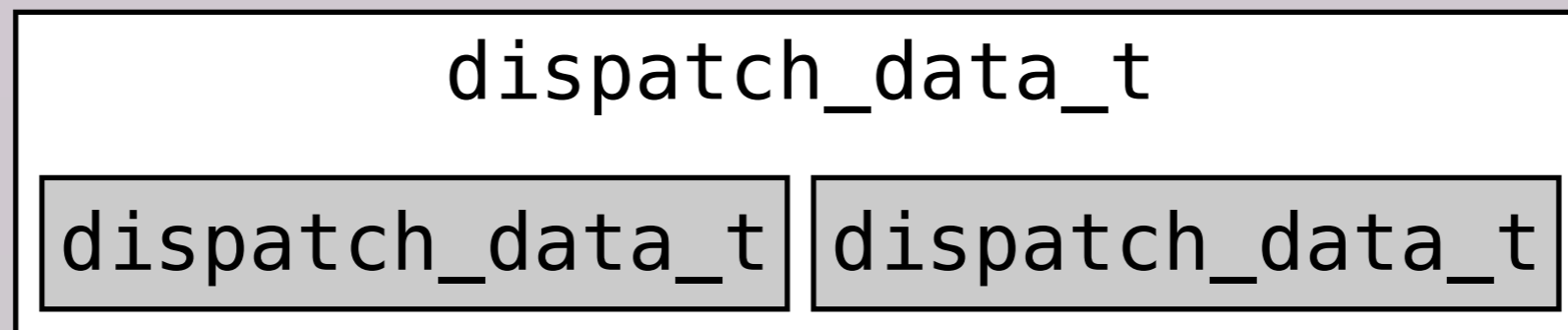
Concatenating Data

```
dispatch_data_create_concat( )
```



Concatenating Data

```
dispatch_data_create_concat( )
```



- References, rather than copies the raw bytes

“Alright, neat, but how does it affect me?”

dispatch_data

dispatch_data

- Toll-free bridged with **NSData**
 - `dispatch_data_t` can be freely cast to **NSData** *
 - But not vice versa

dispatch_data

- Toll-free bridged with **NSData**
 - `dispatch_data_t` can be freely cast to **NSData** *
 - But not vice versa
- **NSData** instances may now contain non-contiguous bytes

dispatch_data

- Toll-free bridged with **NSData**
 - `dispatch_data_t` can be freely cast to **NSData** *
 - But not vice versa
- **NSData** instances may now contain non-contiguous bytes
- – `[NSData bytes]` might have to make a copy of the bytes for you

dispatch_data

- Toll-free bridged with `NSData`
 - `dispatch_data_t` can be freely cast to `NSData *`
 - But not vice versa
- `NSData` instances may now contain non-contiguous bytes
- `-[NSData bytes]` might have to make a copy of the bytes for you
- Switch from `-bytes` to `-enumerateByteRangesUsingBlock:`

Memory Management

- When using ARC, still have to drop down to manual reference counting for CoreFoundation objects sometimes:
 - `CFRetain()`
 - `CFRelease()`

Memory Management

- When using ARC, still have to drop down to manual reference counting for CoreFoundation objects sometimes:
 - `CFRetain()`
 - `CFRelease()`
 - `CFAutorelease()`

CFStream scheduling

- So far have been scheduled on run loops
 - Convenient on the main thread
 - More of a hassle on other threads; likely have to run the run loop yourself

CFStream scheduling

- So far have been scheduled on run loops
 - Convenient on the main thread
 - More of a hassle on other threads; likely have to run the run loop yourself
- `CFReadStreamSetDispatchQueue()` and `CFWriteStreamSetDispatchQueue()`

CFStream scheduling

- So far have been scheduled on run loops
 - Convenient on the main thread
 - More of a hassle on other threads; likely have to run the run loop yourself
- `CFReadStreamSetDispatchQueue()` and `CFWriteStreamSetDispatchQueue()`
- Should work for `NSInputStream` And `NSOutputStream` too thanks to toll-free bridging

Base64

Base64

- New methods on **NSData** for Base64 encoding and decoding

Base64

- New methods on **NSData** for Base64 encoding and decoding
- But you knew that already, right?

Base64

- New methods on **NSData** for Base64 encoding and decoding
- But you knew that already, right?
- Rudimentary support is available back to OS X 10.6 / iOS 4
 - `(id)initWithBase64Encoding:(NSString *)base64String;`
 - `(NSString *)base64Encoding;`

Sorting and Filtering NSMutableArray

`-sortedArrayUsingDescriptors:`

`-sortUsingDescriptors:`

`-filteredOrderedSetUsingPredicate:`

`-filterUsingPredicate:`

- Been there as long as `NSMutableOrderedSet` itself, but only just exposed

NSSecureCoding

- Extension to `NSCoding`
- Introduced in OS X 10.8 / iOS 6
- Protects against corrupt or maliciously crafted archives
 - `decodeObjectOfClass:forKey:`
 - `decodeObjectOfClasses:forKey:`
- Only protects against class changes

NSSecureCoding

- Deeper checking through `-allowEvaluation` on classes:
 - `NSSortDescriptor`
 - `NSPredicate`
 - `NSExpression`
- Check the key paths etc. of an object to check they're suitable for your app, then call `-allowEvaluation`

NSSecureCoding — General Advice

1. Adopt instead of raw `NSCoding`

2. Tell decoder to use it:

- `[NSKeyedUnarchiver setRequiresSecureCoding:YES]`

Doesn't have to be done at the same time as 1.

CGPathCreateWithRoundedRect()

- How did it take this long?!

CGImageSource workflow

- On worker thread:
 - Create CGImageSource
 - Retrieve CGImage
 - Draw into a small bitmap context to force decompression
- On main thread
 - Draw into desired context

kCGImageSourceShouldCacheImmediately

- Pass as a flag when retrieving a **CGImage**
- Should force decompression without all that tedious mucking about in ~~hyperspace~~ bitmap contexts
- *Test and Measure*; don't use blindly!

Bonus Hipster Quiz Question

What is the only public class in the Cocoa frameworks that has no methods of its own?

Bonus Hipster Quiz Question

“you wouldn’t have heard of it...”

What is the only public class in the Cocoa frameworks that has no methods of its own?

