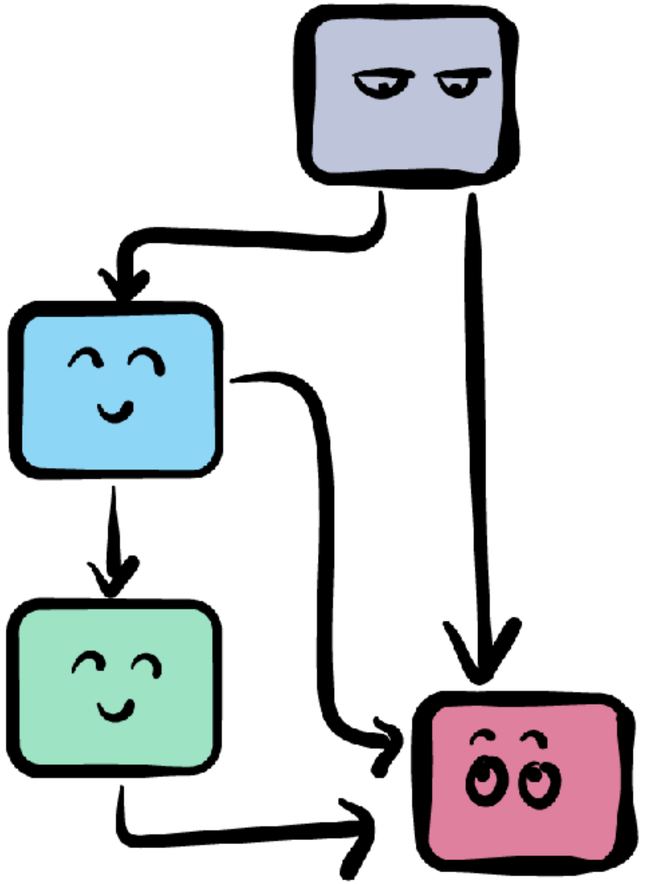
iOS Concurrency with GCD & Operations



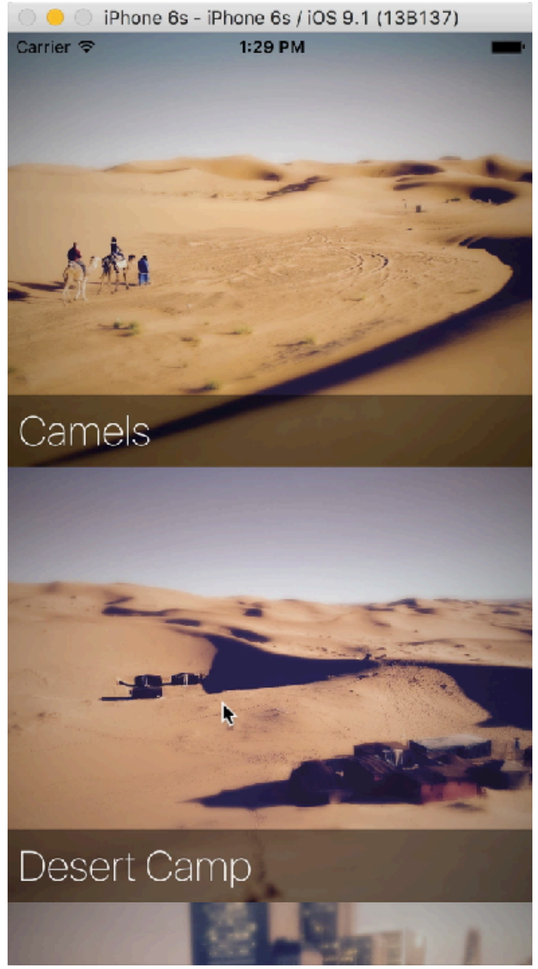
Part 12: Conclusion

Grand Central Dispatch

- Grand Central Dispatch
  - Tasks
  - Queues: serial, concurrent
  - Quality of service
  - Dispatch tasks async, sync
- Dispatch Groups

Operation & OperationQueue

- Operation
- BlockOperation
- OperationQueue
- AsyncOperation
- Dependencies
- Cancelling tasks



GCD vs OperationQueue

Feature	GCD	OperationQueue
All-Finished	✓	don't block main queue!
Dependencies	Chains in same queue only	✓
Barrier	✓	possible but kludgy
Cancel All	DispatchWorkItem only	✓

GCD vs OperationQueue

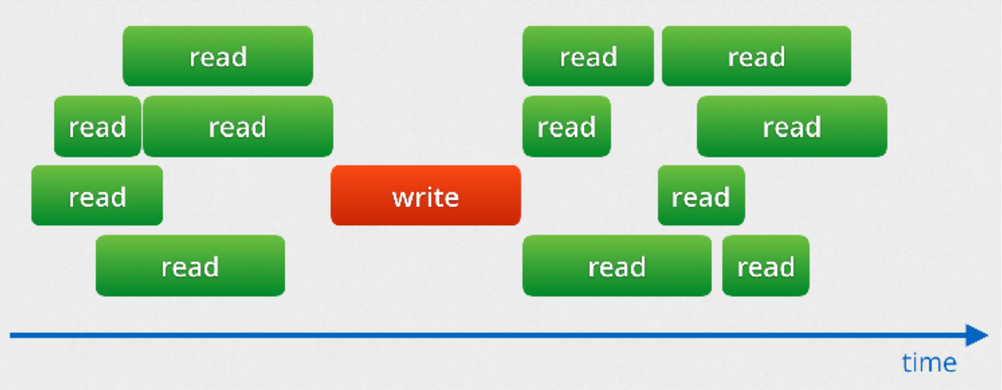
- Async wrapper for sync function
  - Operation encapsulates data into structured, repeatable task
  - Operation provides KVO, suspend-resume
  - GCD is faster, especially for small, simple tasks
- Use Operations for
  - login sequence, CoreData loading, parsing, modal views, observing network
  - One OperationQueue for each view controller

GCD

- When using dispatch queues:
  - Serial: protect resource; synchronize key behavior
  - Concurrent queue for each subsystem with independent data flow (network, DB, image proc);
  - Or a private queue for a specific QoS

Concurrency Solutions

- Thread Safety with Barriers
  - Protect critical section with a dispatch_barrier
  - Making a thread safe object
- Thread Safety with sync()
- Xcode's TSan
- Priority Promotion
  - Prevent priority inversion



Asynchronous Design

- Define your app's expected behavior
  - Tasks, mutable data, dependencies
- Factor out executable units of work
  - Encapsulate in closures or operations; no task is too small
- Identify the queues you need
  - Serial / concurrent / dependencies?
- Improve efficiency

Where to go from here?

Grand Central Dispatch Tutorial for Swift: Part 1/2

Bjørn Ruud on January 7, 2015

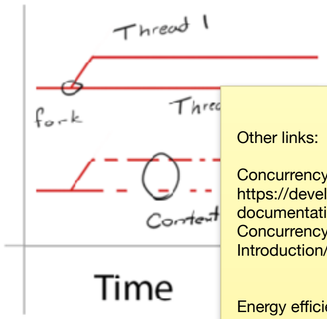
Tweet ถูกใจ 64

Update April 10, 2015: Updated for Xcode 6.3 / Swift 1.2

Update note: This tutorial was updated for iOS 8 and Swift by Bjørn Ruud. Original post by Tutorial Team member Derek Selander.

Although **Grand Central Dispatch** (or GCD for short) has been around for a while, not everyone knows how to get the most out of it. This is understandable; concurrency is tricky, and GCD's C-based API can seem like a set of pointy corners poking into the smooth world of Swift.

In this two-part series, you'll learn the ins and outs of GCD. This first part will explain what GCD does and showcase several of the more basic GCD functions. In the second part, you'll learn several of the more advanced functions GCD has to offer.



Learn about concurrer Grand Central Dispatc tutorial series.

Getting Started

NSOperation and NSOperationQueue Tutorial in Swift

Richard Turton on October 7, 2014

Tweet ถูกใจ 78

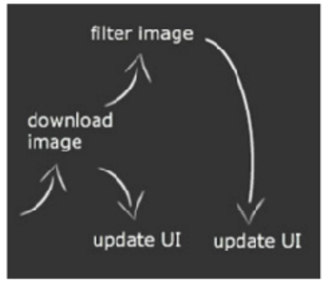
Update 17 April 2015: Updated for Xcode 6.3 and Swift 1.2

Update note: This tutorial was updated to iOS 8, Xcode 6.1 ift by Richard Turton. Original post by Tutorial Team er Soheil Azarpour.

ne has had the frustrating experience of tapping a but- entering some text in an iOS or Mac app, when all of a – WHAM, the user interface stops being responsive.

Mac, your users get to stare at the hourglass or the l wheel rotating for a while until they can interact with again. In an iOS app, users expect apps to respond im- ely to their touches. Unresponsive apps feel clunky w, and usually receive bad reviews.

g your app responsive is easier said than done. Once your app needs to perform more than a l of tasks, things get complicated quickly. There isn't much time to perform heavy work in the un loop and still provide a responsive UI.



Learn how to use NSOperations in your app!

Other links:

Concurrency guide:
https://developer.apple.com/library/ios/documentation/General/Conceptual/ConcurrencyProgrammingGuide/Introduction/Introduction.html

Energy efficiency guide:

https://developer.apple.com/library/ios/documentation/Performance/Conceptual/EnergyGuide-iOS/index.html#//apple_ref/doc/uid/TP40015243-CH3-SW1

WWDC Video:

https://developer.apple.com/videos/play/wwdc2015-226/