

Introduzione

La realizzazione della prima edizione di *Sviluppare applicazioni con iPhone SDK* ci rese consapevoli che non avevamo scritto qualcosa di simile a un libro di programmazione in C, destinato a restare immutato per decenni. Era troppo chiedere che potesse rimanere in libreria per un paio di anni?

Sembra proprio di sì! E non c'era di che lamentarsi. Dall'uscita del primo libro, nei primi mesi del 2009, la piattaforma ha visto crescere la propria diffusione e importanza. In questi anni si è assistito all'introduzione di famiglie completamente nuove di dispositivi, sono state vendute milioni di unità e la piattaforma ha cambiato nome, passando da *iPhone OS* a *iOS* per riflettere meglio le diverse potenzialità e probabilmente per lasciare aperta la porta allo sviluppo futuro di dispositivi completamente nuovi.

Anche gli strumenti SDK sono cambiati profondamente, sono state aggiunte nuove funzionalità, nuovi framework e nuovi strumenti. Apple ha modificato i compilatori e ha rivisto radicalmente Xcode, l'ambiente di sviluppo primario della piattaforma iOS. Il nostro lavoro aveva ogni giorno a che fare con nuove funzionalità, e ogni volta che tornavamo al libro ci rendevamo conto che una parte sempre più significativa del suo contenuto doveva essere aggiornato. Le schermate Xcode, i callback a selettori al posto dei blocchi, l'esposizione di variabili private e i nomi dei metodi nelle header pubbliche: tutti questi elementi non venivano più impiegati e rimandavano a un tempo passato... al 2009.

Un nuovo inizio e nuove possibilità

Chi ha letto il libro precedente e ora si trova a sfogliare le pagine di questo volume non può fare a meno di notare che nulla è stato copiato. Il libro è nuovo al cento per cento. L'esame delle novità introdotte nella piattaforma, legate a Xcode 4, iOS 6 e iPad, ha fatto ritenere le modifiche talmente numerose al punto che era meglio ripartire da zero. In questo modo è stato possibile tenere conto di tutte le novità e di tagliare completamente rispetto al passato, con l'obiettivo di produrre un libro veramente aggiornato.

E di novità ce ne sono davvero tante! Xcode 4 è stato rivisto completamente ed è la prima versione di uno strumento di sviluppo costruito appositamente per iOS. Non si tratta più di concetti per iPhone inclusi in un ambiente IDE per Mac. Xcode 4 ha rivisto completamente il modo in cui gli sviluppatori lavorano con i propri progetti e le

sezioni concettuali rendono più intuitiva l'individuazione delle diverse funzionalità. In fase di preparazione del codice, il nuovo e rivoluzionario strumento Automatic Reference Counting libera gli sviluppatori dalla necessità di includere manualmente le chiamate `retain` e `release`, un lavoro di routine che può provocare problemi di memoria oppure crash di sistema se gestiti in modo non corretto. L'introduzione di processori multicore, per esempio A5, richiede l'adozione di tecniche di programmazione concorrente, con soluzioni offerte da iOS SDK tramite Grand Central Dispatch, una tecnologia che consente ai programmatori di distribuire piccole porzioni di codice e di dati in forma di "blocchi", lasciando al sistema la decisione che riguarda la loro modalità di esecuzione. In effetti è possibile sfruttare al massimo le potenzialità del sistema, come si può notare studiando iOS 6 SDK. Nella prima edizione abbiamo illustrato la maggior parte delle soluzioni che la piattaforma permetteva di adottare e abbiamo visto crescere il libro dalle 250 pagine iniziali fino a quasi 600, facendo saltare scadenza dopo scadenza. E tutto questo facendo riferimento solo a iPhone OS 3. Per riportare allo stesso modo tutte le modifiche introdotte dalle versioni successive di iOS 4 e 6 si dovrebbero scrivere migliaia di pagine, una soluzione decisamente poco "pragmatica".

Si è deciso pertanto di rivedere il focus di questa edizione del libro, che si pone l'obiettivo di mettere il lettore sulla giusta strada, in modo che possa comprendere gli elementi fondamentali e acquisisca familiarità con gli strumenti e con i concetti di base per sviluppare buone pratiche di programmazione. In particolare ci si è concentrati sull'ultimo aspetto della questione, esaminando il genere di cose che non siano solo classi utili o trucchi di compilazione, ma che rappresentino al contrario valori e routine che contribuiscono a realizzare app migliori. Da questo punto di vista si dedicherà molto spazio ad argomenti quali internazionalizzazione, testing, debugging e gestione del codice sorgente. Verranno inoltre adottate tecniche moderne di sviluppo iOS, per esempio l'utilizzo esclusivo di proprietà Objective-C al posto delle variabili istanza tradizionali e dell'impiego di metodi privati in file header pubblici.

Il libro ha lo scopo dichiarato di costituire un prerequisito per gli altri titoli di Pragmatic che si occupano di iOS, per esempio *iOS Recipes: Tips and Tricks for Awesome iPhone and iPad Apps [WD11]* di Paul Warren e Matt Drance, oppure *Core Data [Zar12]* di Marcus S. Zarra; deve quindi fornire i fondamenti che permettano di affrontare lo studio di testi che approfondiranno i molti framework del sistema iOS.

Ancora più importante, al termine del libro il lettore dovrebbe aver raggiunto una padronanza sostanziale delle API iOS indispensabili, ovvero del framework UIKit GUI e delle utility principali del framework Foundation, cui si deve aggiungere la capacità di individuare gli elementi necessari per il lavoro di sviluppo e di affrontare la documentazione di funzionalità che si ritengono utili da implementare.

Contenuto del libro

Vediamo allora il modo in cui si è pensato di raggiungere gli obiettivi prefissati. Inizialmente si scaricherà Xcode dall'App Store, per arrivare infine alla possibilità di caricare applicazioni in iOS App Store. Di seguito è riportata la *road map* delle prossime pagine del libro.

- Il Capitolo 1, *Tweet di benvenuto in iOS 6*, inizia con il download e l'installazione dell'SDK per passare immediatamente all'elaborazione di una prima app, che utilizza

il nuovo framework Social di iOS per inviare un tweet che segnali che il nostro lavoro si è messo in moto. Gli strumenti visuali di Xcode permetteranno di costruire un'interfaccia utente da collegare al primo esempio di codice.

- Il Capitolo 2, *Programmazione in iOS*, affronta le specifiche della codifica introducendo il linguaggio di programmazione Objective-C e i due framework più diffusi nelle app iOS: Foundation e UIKit.
- Il Capitolo 3, *Programmazione asincrona e concorrente*, esamina come e quando si esegue il codice, illustrando le API iOS che utilizzano i callback asincroni e impiegando il sistema Grand Central Dispatch per gestire l'esecuzione del codice in modalità *concurrent*.
- Il Capitolo 4, *I view controller*, torna a occuparsi dell'interfaccia utente per studiare la costruzione di app iOS basate su *Model-View-Controller* (MVC), allo scopo di ottenere un codice più robusto e semplice da gestire.
- Il Capitolo 5, *Viste tabella*, prosegue la costruzione degli elementi dell'interfaccia utente esaminando la vista Table, versatile e ampiamente sfruttata, ovvero il perno della maggior parte delle app per iPhone che devono presentare elenchi di informazioni.
- Il Capitolo 6, *Storyboard e contentainer controller*, insegna a costruire una road map visuale delle schermate che compongono un'applicazione; si vedrà inoltre come generare automaticamente buona parte della logica di navigazione e di presentazione dei dati.
- Il Capitolo 7, *Documenti e iCloud*, fornisce gli strumenti necessari per salvare il lavoro dell'utente nel filesystem e nel nuovo servizio iCloud di Apple.
- Il Capitolo 8, *Disegno e animazione*, introduce l'uso di elementi grafici grazie al framework Core Graphics che disegna immagini e forme, mentre Core Animation dà vita agli oggetti.
- Il Capitolo 9, *Test e correzione delle app*, si occupa delle situazioni che possono andare storte quando si costruisce e si esegue un'applicazione; verranno illustrati gli strumenti SDK di correzione degli errori e di controllo del corretto funzionamento dell'app.
- Il Capitolo 10, *L'App Store e oltre*, conclude il percorso di studio passando dalla fase di apprendimento a quella prettamente esecutiva. Si inizierà mantenendo il codice lungo i passaggi successivi, si eseguirà l'applicazione su un device, la si invierà allo store (dove non verrà rifiutata) e si vedrà come gestire l'app quando finisce nelle mani degli utenti.

Aspettative e requisiti tecnici

In generale, i requisiti tecnici per lo sviluppo in ambiente iOS sono abbastanza ridotti: è necessario avere un computer Mac ragionevolmente nuovo, in grado di eseguire la versione più recente di Mac OS X. I numeri specifici delle versioni cambiano di continuo; verificate l'ultima versione di Xcode presente nell'App Store per conoscere i requisiti tecnici più aggiornati. In questo libro si fa riferimento a Xcode 4.5 e iOS 6 SDK (incluso in Xcode 4.5), eseguiti in ambiente Mountain Lion (10.8.2).

È opportuno che i lettori di questo libro siano esperti programmatori in almeno un linguaggio orientato agli oggetti. Il linguaggio può essere un diretto discendente di C (C++, C# o Java) oppure un linguaggio di scripting orientato agli oggetti, per esempio

Ruby o Python. Nella versione precedente si poneva come requisito la padronanza del linguaggio C e dei suoi concetti di gestione della memoria (puntatori, `malloc()` e così via), ma si è riscontrato che molti lettori non ne erano a conoscenza. In questo libro è presente un'appendice dedicata a coloro che non hanno mai affrontato questi argomenti. Se non avete familiarità con gli operatori di memoria * e &, oppure vi incutono soggezione, l'Appendice A vi aiuterà a conoscere le basi del linguaggio C necessari per lavorare in Objective-C, il linguaggio fondamentale per lo sviluppo di applicazioni iOS.

Risorse online

Questo libro non è fatto solo di pagine o schermate statiche, ma si accompagna a una pagina web sul sito dell'editore originale (in lingua inglese), <http://www.pragprog.com/titles/adios>, dove potete apprendere e accedere a molte risorse online che vi possono risultare utili.

- Scaricate il codice sorgente completo di tutti gli esempi di codice illustrati nel libro, riportati come progetti Xcode pronti all'uso. Una copia degli archivi di codice è disponibile anche sul sito di Apogeo all'indirizzo <http://bit.ly/apo-ios>.
- Partecipate al forum di discussione con altri lettori, con sviluppatori e con gli autori del libro.
- Collaborate a migliorare il libro segnalando correzioni da apportare, legate per esempio a suggerimenti di modifica del contenuto o a refusi.

Al lavoro!

È tutto pronto per iniziare a esplorare l'SDK per iOS 6. Nel prossimo capitolo conoscerete l'ambiente di sviluppo e scriverete una prima app. L'applicazione verrà perfezionata nei capitoli successivi, non appena sarete in grado di aggiungere funzionalità sempre più sofisticate.

Ogni volta che ritenete di avere un problema, consultate il codice sorgente fornito nelle pagine del libro oppure collegatevi ai forum per segnalarci le vostre opinioni. Andiamo a iniziare!