

Apple iOS 8 Overview

DSI Mobile Development Series

Prepared By:



Christopher Duda
Manager Enterprise and Mobile Solutions
June 2014

A. Introduction

This is an exciting time in the mobile development technology space. Recently, Apple released the details for the next version of their mobile computing platform, iOS 8. With over 4,000 new APIs and core features, this platform update is the largest since Apple introduced iOS 2.0 and their heralded App Store. Regardless of your application focus, this new release will change how you design, build, and deploy consumer, enterprise, gaming, and utility oriented mobile applications.

This document will focus on providing a summary (and a relative smattering) of iOS 8's new features and the importance they will have on your current and future development projects.

B. What's New For All Developers

While iOS 8 has over 4,000 new APIs, this section will just focus on the top five truly innovative and roadmap altering features that any Apple-focused mobile development team should start looking into.

1. **Swift Programming Language.** Since the dawn of iOS, Objective-C has been Apple's only available native language. To be brutally honest, Objective-C is one tough language to get your arms around, let alone master (Objective-C is a relict from Apple's 1996 acquisition of NeXT). But that has all changed. Apple developers now have a new native language to build iOS and Mac OSX applications...Swift.

Swift is an innovative new programming language for Cocoa (OSX) and Cocoa Touch (iOS). Developing in Swift is an interactive and liberating experience from Objective-C. The syntax is concise yet expressive, and apps run lightning-fast. Writing Swift code within an Xcode 6 playground (which is something else that is new) shows instant results, while finished apps are compiled into high-performance native machine code. Swift modules can be seamlessly incorporated into existing Objective-C applications and when compiled they will run on iOS 7 and iOS 8 mobile devices. Given that Swift is a new language,

development teams should start the process of educating themselves. DSI is updating our Mobile Development Best Practices to include Swift-based development.

- 2. Extensions.** This is perhaps the most requested feature by developers within the iOS ecosystem. Now with iOS 8, developers will be able to create extensions that allow applications to securely communicate, share data, and share processes directly with one another. Using this new API set effectively extends an app's sandbox security wall to include a short-lived tunnel to another application. Once the tunnel is created, the third-party application can share data, processes, and even inject user experience components (via remote views) within the host application. During this interaction, the host app's data and settings are never exposed to the third-party application. Also, the device's user is in complete control to either allow or deny the connection as they see fit. The extension tunnel executes out-of-process with the host app and is always sandboxed, meaning any unexpected issues are contained and not exposed to the rest of the mobile device. This feature will fundamentally change the Apple iOS app landscape from a pallet of walled app-based silos to a collaborative canvas, where vendors can provide discrete services and features to any number usage scenarios.
- 3. Universal Storyboard.** Since the inception of storyboards, Apple has advocated modeling and managing the user experience using this powerful and intuitive design tool. Given that different screen sizes and device orientations demand a different user experience, developers and designers often created multiple storyboards to manage the user various experience options. This meant that additional time and resources were needed to design, implement, and support these options. Development teams often had to weigh the urgency to support multiple orientations or platforms with other central application features.

With Xcode 6, designers and developers now have access to a universal storyboard, on which all UX orientations and platforms can be modeled using a single set of Interface Builder views and components. Using new "Sizing Classes" and adaptive UI constraints, a

single storyboard can be used to manage, design, and consolidate the UX for any number of screen sizes and orientations. Additionally, the Xcode 6 storyboard preview assistant allows the UI to be viewed as if it was being rendered on a specific device and orientation. This new toolset will allow development teams to fully embrace and support the complete range of current UX options, as well as any new UX choices (such as the rumored forthcoming iOS side-by-side multi-tasking model) or hardware changes (such as the rumored larger screen iPhone 6 or the rumored larger iPad Pro tablet). Regardless of what future features or devices become available, Xcode 6 universal storyboards will reduce UX resource and delivery times on your current projects.

4. **Notification Center Widgets / Interactive Notifications.** Another feature often requested was the ability to provide custom functionality and features to the iOS notification center, as well as provide interactive prompts/actions to the iOS alerts. With iOS 8 both capabilities will be available.

The new iOS 8 capability to add widgets will allow apps to provide simple and helpful utilities to the notification center panel. Users will be allowed to control which widgets are displayed, as well as access a set of vendor provided settings to control their interaction with the widget. The likely usage scenario for the widgets will be to provide timely and convenient access to your app's most central and user critical data and processes. However, widgets could also be used to provide supplemental information or provide the ability to quick launch into the application.

The new interactive notifications API will allow the user to be presented with and process custom actions within the notification area and on the lock screen. This will allow the user to provide instant and secure interaction with an app's workflow without having to switch the user's context to the target app. Scenarios where users can reply to a message, accept an application prompt, act on an event, or even provide customized user input will now be possible.

5. **CloudKit.** It seems that cloud presence and features are everywhere within IT. For mobile applications, the cloud is more than a buzzword. It is a central component for providing off-device features and data. With iOS 8, Apple is releasing CloudKit, a SDK and service combination will allow any iOS application to seamlessly maintain a cloud presence. With CloudKit, Apple will provide server-based services and interfaces, as well as, provide the app-side interface logic components. CloudKit apps can quickly have a free, fully managed and hosted online presence that supports:
- Cloud-based SQL databases (up to 10 terabyte in size)
 - Asset libraries (up to 1 petabyte in size)
 - Full user subscription service with alerts
 - Comprehensive developer portal for management and analytics

Apple is aware that developers have access to other cloud platforms, and they fully support communication and interaction with these platforms within iOS. However, CloudKit will provide developers with another cloud-based solution to leverage with perhaps the lowest cost (free – unless your needs are unusually large) and the easiest integration footprint (build into iOS and OSX).

6. **Xcode 6.** *I know this section was to highlight five new iOS platform features, but there is another that should be emphasized - so consider this a bonus!* Apple's Xcode development tools suite has been updated to coincide with the introduction of iOS 8 and OSX Yosemite. Hundreds of new features and redesigned components can be found in Xcode 6. A few of these new features have been mentioned in this document (e.g. Swift language support, playgrounds, universal storyboards) and it would be impossible to highlight, or to do justice, to all of the other changes. However, a few of the other new features that any mobile development team should look to leverage include:
- Universal document find and replace (even in Interface Builder)
 - New localization support leveraging XLIFF
 - Real-time rendering of custom controls in Interface Builder
 - New debugging options (view debugging, queue debugging, new debug gauges)

- New testing options (asynchronous testing, performance testing, profile guided optimization, updated Xcode Bots)
- Structured beta testing through the App Store using TestFlight
- Redesigned and optimized Instruments application
- New SpriteKit and SceneKit design tools
- Generation of all iOS launch images (at all supported resolutions)
- New asset catalog support

C. What's new For Enterprise-Focused Developers

Quietly, but steadily, Apple's iOS platform has become the premiere mobile platform for enterprises. 98% of the Fortune 500 actively deploy iOS products within their mobile infrastructures. This trend has been no accident. Apple has continuously added robust and advanced enterprise support into the iOS platform. Comprehensive MDM support, enterprise single sign on, world-class internal and third-party data protection/encryption, and per-app VPN support are just several of Apple's often-heralded enterprise features. Apple's iOS 8 will help extend their enterprise lead. The platform will now include:

- Passcode-protection of all the major data types (Calendar, Contacts, Mail, Messages, Notes, Reminders) and third-party apps
- S/MIME controls per message
- VIP Threads to allow a user to subscribe to important email threads and get notifications on their lock screen, without having to designate the sender(s) as "VIPs"
- Optimized and enhanced email and calendar navigation and handling
- Easier access to corporate documents
- Enhanced MDM support.

However, this document is meant to have a development focus, so we have included a few of the new API's that enterprise developers will love.

1. **Local Authentication.** Although this feature will be of interest to most mobile development teams that produces an app that includes user sign on, DSI expects that enterprise developers will quickly adopt this. On capable hardware – such as the iPhone 5s, iPhone 6 (rumored),

iPad Air 2 (rumored), and the iPad Mini Retina 2 (rumored) – developers can now have access to Touch ID biometric fingerprint authentication from within apps. iOS 8 exposes a simple API that tells the host application whether the authentication succeeded or not. At no time does the host app have access to the user's biometric information. In fact, Apple's own applications cannot even access it. The user's biometric information stays locked away in the processor's secure enclave. Allowing Touch ID access will provide a greater level of security to third-party apps and allow for new authentication-based workflows to be created (including action sign off and approval scenarios).

2. **Extensions / Notification Center Widgets / Interactive**

Notifications. These new features were covered above, but for enterprise developers we think they deserve to be called-out again. These three features will allow enterprise developers to create app and functionality suites that previously have only been available on desktop or web-based solutions. Each will allow strategic and secure access of critical line-of-business data and processes between applications, as well as lower the interface footprint for enterprise users that often have to focus on multiple tasks at a time. Additionally, all of these new extension points and notification actions can be configured and managed through the MDM interfaces.

- 3. Document Providers.** Apple iOS does not have an exposed file system at the user-level. Although lamented by some, this sandboxed, security-focused limitation has made iOS very resistant against malware and other related breaches. Starting with iOS 8, Apple will allow applications to leverage third-party document providers. This new API will allow applications to access document-based assets (including JSON, Binary/BLOB, XML, or vendor formatted data) that reside on off-device services. The third-party document service providers - which will initially include Apple iCloud, Microsoft OneDrive, Box.net, WebDav, and Drop Box (rumored, but highly expected) - will register themselves directly with iOS. At runtime, iOS will dynamically extend your app's sandbox security wall to include any selected document accessed from one of these services. When the document is no longer needed, the sandbox will automatically be constricted to

no longer include the resource. All of the settings and permissions related to the document providers will be managed by the user at the iOS level. This new feature will allow enterprise apps (and consumer ones too) to securely and consistently source and share information through established document service providers with very little application overhead or management. This will open the door for new types of application workflows that allow mobile users to seamlessly access document-based data within iOS.

D. What's new For Gaming And Media-Focused Developers

Not be left out, gaming and media-focused developed will find a plethora new features and APIs available in iOS 8. Like with the discussion of Xcode 6 above, it would impossible to do justice to all of the new options available (and it would be also be a bit out-of-focus with the rest of this document). However, for those interested, here is a quick rundown of the major new features iOS 8 provides for gaming and media-focused development teams:

- Metal API - This will provide low-level access to the device GPU that bypasses the normal application and driver stack often yielding 5-10x frame rendering speed improvements.
- Many new SpriteKit enhancements.
- New SceneKit SDK that will bring 3D rendering in a similar fashion as SpriteKit.
- New SpriteKit and SceneKit design and debugging tools.
- New non-destructive photo/video editing extensions.
- Manual control of camera settings (e.g. focus, white-balance, ISO, shutter speed).
- Full access to photo/video library regardless of device or iCloud source location.

E. Conclusion

DSI is excited about Apple's iOS 8 feature and platform announcements. For over 20 years, DSI has been a pioneer in building mobile solutions for both the enterprise and consumer marketplaces. We have already started reworking our own Mobile Development Best Practices and our related

technical skillsets to include iOS 8, Swift, and Xcode 6. If you have a mobile development team or a mobile-focused project, we would welcome the opportunity to talk to you and explore how Apple's iOS 8 mobile platform can be leveraged and mastered.