

iOS SDK Release Notes for iOS 7 Seed 3

Important: This is a preliminary document for an API or technology in development. Although this document has been reviewed for technical accuracy, it is not final. This Apple confidential information is for use only by registered members of the applicable Apple Developer program. Apple is supplying this confidential information to help you plan for the adoption of the technologies and programming interfaces described herein. This information is subject to change, and software implemented according to this document should be tested with final operating system software and final documentation. Newer versions of this document may be provided with future seeds of the API or technology.

Contents:

- Introduction
- Bug Reporting
- Notes and Known Issues

Introduction

iOS SDK 7.0 provides support for developing iOS apps, and it includes the complete set of Xcode tools, compilers, and frameworks for creating apps for iOS and OS X. These tools include the Xcode IDE and the Instruments analysis tool, among many others.

With this software you can develop apps for iPhone, iPad, or iPod touch running iOS 7. You can also test your apps using the included iOS Simulator, which supports iOS 7. iOS SDK 7.0 requires a Mac computer running OS X v10.8.4 (Mountain Lion) or later.

This version of iOS is intended for installation only on devices registered with the Apple Developer Program. Attempting to install this version of iOS in an unauthorized manner could put your device in an unusable state.

For more information and additional support resources, visit <http://developer.apple.com/programs/ios/>.

Bug Reporting

To report any bugs not mentioned in the Notes and Known Issues section, use the Apple Bug Reporter on the Apple Developer website (<http://developer.apple.com/bugreporter/>). Additionally, you may discuss these issues and iOS SDK 7.0 in the Apple Developer Forums: <http://devforums.apple.com>. To get more information about iCloud for Developers, go to <http://developer.apple.com/icloud>.

Notes and Known Issues

The following issues relate to using iOS SDK 7.0 to develop code.

AirDrop

Notes

- AirDrop in iOS 7 Seed 2 or later is not compatible with AirDrop in iOS 7 Seed 1.
- Contacts Only support requires that:
 - You are logged in to iCloud.
 - The sender and receiver have each other's Apple ID email addresses in Contacts on their devices.

Known Issues

- Contact Only currently requires that you have an active Internet connection.
- The App Store should support the ability to find an app capable of opening a received document based on the document's Uniform Type Identifier (UTI). Currently, just a blank screen is displayed.

AirPlay

Fixed in Seed 3

- When sending music to an Apple TV, advancing to the next track may fail.
- Audio volume may increase dramatically and unexpectedly during AirPlay.

AV Foundation

Notes

`AVCaptureSession` now uses your app's `AVAudioSession` by default.

Calendar

Known Issue

- iCloud calendars or reminders may be duplicated after adding new events.
 - Workaround:** Hide the duplicated calendar from view and things will continue to sync normally.
- iCloud calendar events may not sync through fetch, push, or manual refresh as expected.
 - Workaround:** Create a new event or edit an existing event to trigger the sync.

Car Display

Known Issues

Support for Car Display will be provided once compatible systems are available.

Contacts

Known Issues

Some high-resolution photos will not sync (via iCloud and other services) when set as a Contact's Caller ID photo.

Workaround: Use a lower-resolution photo.

Core Animation

Notes

When the `UIViewGroupOpacity` key is not present, the default value is now YES. The default was previously NO.

This means that subviews of a transparent view will first be composited onto that transparent view, then the precomposed subtree will be drawn as a whole onto the background. A NO setting results in less expensive, but also less accurate, compositing: each view in the transparent subtree is composited onto what's underneath it, according to the parent's opacity, in the normal painter's algorithm order.

iCloud

Notes

- iCloud Keychain in iOS 7 Seed 3 is not backward compatible with iOS 7 Seed 2 and OS X Mavericks Seed 2.

If you had iCloud Keychain and/or Keychain Restore enabled from iOS 7 Seed 2, both features are disabled upon updating to iOS 7 Seed 3.

After updating to iOS 7 Seed 3, go to Settings > iCloud > Keychain and reenable iCloud Keychain on your devices.

See *OS X v10.9 Release Notes* for more details.

- Activation Lock, a new feature of Find My iPhone, is turned on automatically when Find My iPhone is enabled on any device running iOS 7. Activation Lock requires users to enter their Apple ID and password to turn off Find My iPhone, sign out of iCloud, erase the device, or reactivate the device after an erase. It's important to do one of the following before transferring a device to a new user:

Sign out of iCloud.

Turn off Activation Lock in Settings > iCloud > Find My iPhone.

Fixed in Seed 3

- The keyboard is not present in the iCloud Keychain approval dialog after screen unlock.
Workaround: Tap on either Don't Allow or Allow. The keyboard will be presented the next time.
- If you change the country associated with a phone number in Settings > iCloud > *Account Name* > Keychain, you will not be able to view the changed country name/code.
Workaround: Turn off Keychain Restore, and then turn it back on.
- If you have "Reset All Contents and Settings" on a device (Settings > General > Reset) that:
 - Has an iCloud account set up
 - Has Find My iPhone Activation Lock enabled
 - Has not previously navigated to Settings > iCloudyou won't be asked to enter your iCloud credentials to disable Activation Lock prior to erasing. After the device is erased, your device will be activation locked.
Workaround: Delete the iCloud account prior to erasing the device. If in this state, enter your iCloud account credentials to unlock the device.
- iCloud Keychain users who set up an additional device will receive an approval request to the other devices with iCloud Keychain turned on. The approval request will not be displayed if their screens are off and the devices are idle.
Workaround: Wake up any peer device to accept the approval request.
- In iOS 7, Find My iPhone and Bookmarks toggles are no longer supported on secondary iCloud accounts. When you update your device with either of these options enabled on a secondary iCloud account, they will be turned off with no user notification.
Workaround: Set up the primary iCloud account via Settings/iCloud and then enable Find My iPhone or Bookmarks.
- On rare occasions, users might experience difficulties setting up iCloud Keychain on some of their devices.
Workaround: A temporary Reset iCloud Keychain button is available in iCloud Settings > Account > Account Security.

Known Issues

- Sometimes when you disable iCloud Keychain, you may not be able to reenableView it.
Workaround: Disable the Keychain on all enabled devices, then reenableView it.
- Keychains are not synced among devices when they are on a wireless network with WEP security.
- When setting up iCloud Keychain in Setup Assistant on iPhone 4 and iPhone 4s, the keyboard covers the iCloud Security Code entry field and the Advanced Options link.
Workaround: Scroll down to see the hidden entries.
- Users in Switzerland, France, Spain, Germany, Poland, Finland, and possibly some other countries, will be unable to restore their keychains using phone number validation.
Workaround: Use Random Security Code.
- Credit cards deleted in Safari Settings > AutoFill are removed from the device but not from other devices.
Workaround: Making another local change (like adding a credit card) will force the sync.
- Users signing out of iCloud by deleting their iCloud account, erasing all contents and settings,

or erasing the device via Find My iPhone and signing back in will not be presented with the correct iCloud Keychain setup screens.

Workaround: Turn on iCloud Keychain, then approve with iCloud Security Code > Forgot Security Code > Reset iCloud Keychain.

- Users who have a single device with iCloud Keychain on will not be presented with the correct iCloud Keychain setup screens after restoring from a backup.

Workaround: Turn on iCloud Keychain, then approve with iCloud Security Code > Forgot Security Code > Reset iCloud Keychain.

- Nonpurchased eBooks and PDFs will not be restored when iCloud-restoring an iOS 7 device with a pre-iOS 7 backup. Purchased content will be restored without issue.

Workaround: Use iTunes sync.

MapKit

Notes

If the calls to `renderInContext` on the layer backing an `MKMapView` are made off the main thread, they should be eliminated or moved to the main thread. Failure to do so can cause an app crash. Instead of using `renderInContext`, use the new `MKMapSnapshotter` APIs.

Messages

Fixed in Seed 3

- On a clean install, launching Messages presents an empty message list instead of a new compose window.
- The existing Messages database may be deleted after a restore or upgrade if:
 - You do not sign in to iCloud during the Setup Assistant.
 - The iMessage and iCloud accounts use different IDs.
- Messages app crashes after viewing any attachments on two separate threads.

Workaround: The attachments are viewable after relaunching Messages.

Multipeer Connectivity

Notes

Many updates to the `MCSession` APIs are included in this seed. The APIs now function as they were presented at WWDC. Major changes and enhancements include:

- The following new method on `MCSession` has been implemented: `- (NSOutputStream *)startStreamWithName:(NSString *)streamName toPeer:(MCPeerID *)peerID error:(NSError **)error`

This delegate method has been implemented: `- (void)session:(MCSession *)session didReceiveStream:(NSInputStream *)stream withName:(NSString *)streamName`

```
fromPeer:(MCPeerID *)peerID
```

- The following new method on `MCSession` has been implemented: – `(NSProgress *)sendResourceAtURL:(NSURL *)resourceURL withName:(NSString *)resourceName toPeer:(MCPeerID *)peerID withCompletionHandler:(void (^)(NSError *error))completionHandler`

The delegate method to start receiving a resource from remote peer has been implemented as:

```
– (void)session:(MCSession *)session didStartReceivingResourceWithName:(NSString *)resourceName fromPeer:(MCPeerID *)peerID withProgress:(NSProgress *)progress
```

The delegate method to finish receiving a resource from remote peer and save the content in a temporary location is implemented as:

```
– (void)session:(MCSession *)session didFinishReceivingResourceWithName:(NSString *)resourceName fromPeer:(MCPeerID *)peerID atURL:(NSURL *)localURL withError:(NSError *)error
```

Note: The app is responsible for moving the file to a permanent location within its sandbox.

- `MCAvertiserAssistant` is a new class. It is a convenience class for implementing an Advertiser, which handles invitations and connections to an `MCSession`.
- `MCSession` has a new `initWithPeer:` method.
- `sendResource` has added “with” to the `completionHandler`.
- The timeout has been removed from `connectPeer:withNearbyConnectionData`.

See “WWDC Session 701: Multipeer Connectivity” for more information on these APIs.

Music Player

Fixed in Seed 3

`MPMediaPickerController` has been disabled for this seed. It will immediately return as if the user canceled the selection.

Networking

Notes

- Cellular fallback is a technology that will attempt to connect over the cellular interface when a connection over the Wi-Fi interface does not succeed. There are a number of cases where connection failures may occur over Wi-Fi, such as a cable modem in need of a reboot or a firewall that blocks access to certain resources.

If your app restricts network operations over the cellular network, verify that your code is using the proper API to implement that policy.

For example, if your app watches `SCNetworkReachability` for the `kSCNetworkReachabilityFlagsIsWWAN` to monitor when Wi-Fi is available, it may not behave correctly under cellular fallback.

If an app uses this method and then attempts to connect, thinking it will get a connection over Wi-Fi, the app may inadvertently end up with a connection over the cellular network. The lack of the `kSCNetworkReachabilityFlagsIsWWAN` is not a guarantee that your connection will not go

over cellular. If your code needs to enforce that a connection not go over cellular, it needs to specify that when making a networking request.

Instead, for `CFNetwork` code using `CFReadStream`, `CFHTTPStream`, or similar APIs, set the `kCFStreamPropertyNoCellular` property to true on the CF object before connecting.

For `NSURLRequests`, call `[request setAllowsCellularAccess: false]`.

In addition to making sure your code will never be connected over cellular because of cellular fallback, this call ensures that your code will never be connected over cellular because of the VPN or convoluted routing problems (connecting to IPv6 over cellular because only IPv4 is available over Wi-Fi, etc.).

To verify that your app is behaving properly, you should test against a broken Wi-Fi network. The easiest way to set up a broken Wi-Fi network is to set up an Apple AirPort Base Station or other Wi-Fi access point to hand out a DHCP address and leave the access point disconnected from the Internet. Another simple way of testing is to unplug the cable or DSL line from your cable or DSL modem.

- Two low-level networking APIs that used to return a MAC address now return the fixed value `02:00:00:00:00:00`. The APIs in question are `sysctl (NET_RT_IFLIST)` and `ioctl (SIOCGIFCONF)`. Developers using the value of the MAC address should migrate to identifiers such as `-[UIDevice identifierForVendor]`. This change affects all apps running on iOS 7.
- All `NSURLSession` session tasks now start suspended rather than running. This allows you to set up their state before the task is started.

Newsstand

Fixed in Seed 3

Newsstand background downloads that use HTTP basic or digest authentication will prompt the user with authorization alerts (`NSURLConnectionDelegate` Protocol callbacks for `connectionShouldUseCredentialStorage` and `willSendRequestForAuthenticationChallenge` are not called correctly).

Objective-C Runtime

Notes

Due to changes in how the `isa` field is implemented, `*self` may change during enumeration (for example, if the container is retained).

Workaround: When implementing `countByEnumeratingWithState`, do not set `state->mutationsPtr = self`.

Passbook

Notes

- In previous versions of the iOS SDK, Passbook did not validate the back fields on passes completely. This has been corrected, but it means that some passes added on iOS 6.0, 6.1, and iOS 7.0 Seed 1, will be invalid on iOS 7 Seed 2 and later. The validation rules have not changed, it was just a bug in validation in previous implementations. If you have a pass that you were

previously able to add to Passbook, but no longer can, watch the console log for more than one field that has the key NAME. Field keys need to be unique.

- As of iOS 7 Seed 3, `PKPassLibraryDidCancelAddPasses` is a new status code in the `PKPassLibraryAddPassesStatus` enum. It signifies that the user tapped Cancel in an add-passes alert.

Safari

Known Issues

- Upgrading to iOS may reset Safari's settings to their default values.
- Safari website credentials do not always autofill between mobile and desktop versions.

Workaround: Use the desktop version if the site offers it.

Security

Notes

- `-[UIDevice uniqueIdentifier]` is no longer accepted in submissions to the App Store. In iOS 7, apps that are already on the store or on users' devices that call this removed API will no longer be returned the UDID. Instead, `-[UIDevice uniqueIdentifier]` will return a 40-character string starting with `FFFFFFFF`, followed by the hex value of `-[UIDevice identifierForVendor]`. It is important to consider how this will affect existing apps. Consider submitting updates that no longer access the UDID.
- iOS now requests user consent for apps to use audio input on all iOS 7 devices. For devices sold in China, iOS will also request user consent for apps to use the camera hardware. The operating system will present the consent alert when you set the category of the instantiated `AVAudioSession`. The `AVAudioSessionCategories` that will present the alert are `AVAudioSessionCategoryRecord` and `AVAudioSessionCategoryPlayAndRecord`.
If the user doesn't allow access, the audio session data will be all zeros (silence). For devices where camera access is requested and denied by the user, the video capture session is a black screen.
- The API `gethostuuid()` has been removed and will not be accepted for submission to the store, regardless of the targeted OS. For existing apps running on iOS 7, the function will return a `uuid_t` representation of the vendor identifier (`-[UIDevice identifierForVendor]`).

Settings

Fixed in Seed 3

In General > Background App Refresh, the per-app settings are not honored.

Setup

Fixed in Seed 3

After erasing the device from Settings > General > Reset > Erase All Content & Settings, CrashReporter logs will no longer be copied from the device to the computer.

Workaround: Restore the device from an IPSW instead of using Erase All Content & Settings on the device.

Siri

Notes

This seed includes development versions of new male and female Siri voices for U.S. English and French, and a male voice for German.

Social

Notes

Up through iOS 6, when using `TWTweetComposeViewController` and `SLComposeViewController` (the latter only for Twitter and Weibo, but not Facebook), if the caller supplies a `completionHandler`, the supplied `completionHandler` is responsible for dismissing the view controller. As of iOS 7, if the app links against the iOS 7 SDK, the view controller will dismiss itself even if the caller supplies a `completionHandler`. To avoid this, the caller's `completionHandler` should not dismiss the view controller.

Known Issues

When using the iOS 6.1 SDK on OS X v10.8 Mountain Lion, if you use the iOS 5.0 or iOS 5.1 Legacy SDK in iOS Simulator, you will not be able to use Twitter features: attempting to sign in to Twitter via the Settings pane will fail, and `Twitter.framework` will not work correctly. If you need to test Twitter features, you will need to choose either an iOS 6.1 or iOS 6.0 Simulator run destination, or you can test with iOS 5.x on a device.

Springboard

Notes

- Active touches are no longer canceled when the user takes a screenshot.
- Dynamic wallpaper is not available on iPhone 4.

Fixed in Seed 3

- Passcode Lock and Auto-lock time settings are not always respected. Sometimes the device does not require a passcode when the settings indicate it should.
- Some apps may receive duplicate push notifications or no notification.
- The app switcher does not show all of the suspended apps.

Known Issues

- Passwords with diacritical marks are not supported in this seed. If you have diacritical marks in your password, change it before upgrading to Seed 3 as you will be unable to unlock your device.
- Springboard crashes when accessing some web clips in the App Switcher.
- When restoring from an iCloud backup on a device with hundreds of apps, Springboard may crash repeatedly.

Sprite Kit

Known Issues

`+[AVPlayer playerWithPlayerItem:]` returns an `AVPlayer` object, but the current item will return `nil` if the player was created on another thread.

Sync

Known Issues

The Wi-Fi sync option in iTunes is checked, but Wi-Fi devices do not appear in iTunes.

Workaround: On OS X, install the iOS 7.0 SDK after installing iTunes. Wi-Fi sync is not supported on Windows.

UIKit

Notes

- `+[UIPasteboard pasteboardWithName:create:]` and `+[UIPasteboard pasteboardWithUniqueName]` now unique the given name to allow only those apps in the same application group to access the pasteboard. If the developer attempts to create a pasteboard with a name that already exists and they are not part of the same app suite, they will get their own unique and private pasteboard. Note that this does not affect the system provided pasteboards, general, and find.
- Starting with Seed 2, apps default to using the new view controller-based status bar management system. To opt out of this, add a value of `NO` for the `UIViewControllerBasedStatusBarAppearance` key to your Info.plist.
- When using Auto Layout to position a `UIButton`, if you set the content compression resistance or content hugging priority to minimum, the button will have ambiguous layout.
Workaround: Don't use a content compression resistance or content hugging priority of less than 2 for `UIButton`.
- `UIScenEdgePanGestureRecognizer` allows you to perform actions in response to swipes over the edge of the screen using the same heuristics that the system uses for its own gestures. Use this if you have a navigation semantic of your own that doesn't use `UINavigationController` but should include this gesture (e.g., Safari).

This gesture recognizer has a property that describes the edges on which it's active.

`UIRectEdge` is a new enum type that this property and `-[UIViewController edgesForExtendedLayout]` can share. `UIRectEdge` replaces `UIExtendedEdge`, which will be removed. The members of both have the same values. Use Xcode to replace all instances of

“UIExtendedEdge” in your project with “UIRectEdge”.

- Blurred layers are not available on iPhone 4.
- Parallax is not available on iPhone 4.
- Letterpress text is not available on iPhone 4.

Fixed in Seed 3

- When a device without a passcode is connected to an untrusted computer, iTunes will tell the user to approve on the phone, yet the phone never asks the user to approve. The phone is trusted anyway, so there is no functional impact, but the UI is missing.
- `UISwitchonTintColor` is always green irrespective of what it's set to.

Known Issues

- IB support for `-topLayoutGuide` and `-bottomLayoutGuide` is not functional in this seed.
- `UIPickerView` with custom views doesn't show the views in the selection indicator.
- `UIRefreshControl` and the associated table view are hidden under the navigation bar when `setTranslucent:` is `NO`.
- `UIButtonTypeInfoLight`, `UIButtonTypeInfoDark`, and `UIButtonTypeDetailDisclosure` buttons all look the same.
- The `UIImagePickerController` does not show a live preview when using custom overlays.
- The title of the `UIRefreshControl` appears below the navigation bar.

VoiceOver

Fixed in Seed 3

Reminders does not work with VoiceOver.

Weather

Notes

Weather conditions are not animated on iPhone 4.

WebKit

Notes

- Previously, when the viewport parameters were modified, the old parameters were never discarded. This caused the viewport parameters to be additive.

For example, if you started with `width=device-width` and then changed it to `initial-scale=1.0`, you ended up with a computed viewport of `width=device-width, initial-scale=1.0`.

In iOS 7, this has been addressed. Now you end up with with a computed viewport of `initial-`

scale=1.0.

- Previously, when using `<meta name="viewport" content="initial-scale=1.0, user-scalable=1">`, the scale could be incorrect after rotation.

Now, if a user has not scaled the page explicitly, the page is restored to its initial scale. Also, the current scale is now correctly restricted within the `min-scale`, `max-scale` bounds.

- Some images for Retina display devices appear at twice the size that they should instead of being scaled down 50%. This issue can also occur with any image to be scaled that is larger than its container and that uses either the `background-size` property or the `-webkit-background-size` property. This issue affects both native apps that display web content and web pages viewed in Safari for iPhone.

The `background` CSS shorthand property now resets the value of the `background-size` property to `auto` when background size is not specified. This new behavior is per the CSS spec, and the old behavior is available only for apps linked to an SDK prior to iOS 7.0 when running on iOS 7 or later.

Workaround: Specify the `background-size` property or the `-webkit-background-size` property after the `background` shorthand property in the CSS stylesheet for the web content being displayed.