Best Practices for Building Apps Used in Business and Education

Session 304

David M. O’Rourke iOS Managed Configuration Team
Agenda
Agenda

Modern App Design Practices
Agenda

Modern App Design Practices
Architecture Overview of Shared iPad
Agenda

Modern App Design Practices
Architecture Overview of Shared iPad
Cover What iOS Technologies to Adopt for Shared iPad
Agenda

Modern App Design Practices
Architecture Overview of Shared iPad
Cover What iOS Technologies to Adopt for Shared iPad
New Opportunities to Enhance Your App
Modernize Your Application
Modernize Your Application

Customers expect a fully modern and up to date iOS experience
• Stay current, adopt iOS features…
Modernize Your Application

Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Modernize Your Application

Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Modernize Your Application

Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Modernize Your Application

Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Modernize Your Application

Customers expect a fully modern and up to date iOS experience

• Stay current, adopt iOS features…

Some example technologies:
Even if students share an iPad, they can make it their own.

Shared iPad is a new iOS 9 feature for schools that share devices and still want to provide personalized experiences. Before class begins, you can assign any iPad to any student in your classroom. Students know which devices to use because their pictures are on the Lock screens. And they can access their personal content with a password or an easy-to-remember four-digit PIN. After students log in, their homework, apps, and assignments appear exactly as they left them.
Shared iPad

Even if students share an iPad, they can make it their own.

Shared iPad is a new iOS feature for schools that share devices and still want to provide personalized experiences. When you log in, you can adjust your iPad to reflect your personal style. Students can set their own wallpaper, customize their home screen, and use their own email and files.

With Shared iPad, students can access their personalized content with a personal or school-issued account. Their files and data are backed up and restored across their devices, so they can work on their homework and assignments securely, even when they're at home.
Shared iPad

Introduced with iOS 9.3

• Allows schools to deploy Shared iPad
Shared iPad

Introduced with iOS 9.3

• Allows schools to deploy Shared iPad

Managed Apple ID is the user’s sign-in

• Managed by the organization
Shared iPad

Introduced with iOS 9.3
• Allows schools to deploy Shared iPad
Managed Apple ID is the user’s sign-in
• Managed by the organization
Account data must be cloud based
• Student experience is the same on all Shared iPad
Goodbye, Mia
Shared iPad

All of Mia’s data in the cloud

“Mia” may use a different iPad tomorrow
• Data may be purged by iOS

iOS provides four core cloud technologies

Quick overview of Shared iPad expectations
Shared iPad Behind the Scenes
Shared iPad Behind the Scenes

User switching

• Only one active user
• Sign out to switch to another user
Shared iPad Behind the Scenes

User switching
- Only one active user
- Sign out to switch to another user

Apps see single user device
- Sign out is power down from an app perspective
Shared iPad Behind the Scenes

User switching
- Only one active user
- Sign out to switch to another user

Apps see single user device
- Sign out is power down from an app perspective

User data from the cloud is cached on device
- Improves performance
- Offline for field trips
- Cache data is purged by iOS if necessary
Move Your App Data Into the Cloud
Consistent user experience on different devices
Move Your App Data Into the Cloud
Consistent user experience on different devices

iOS cloud technologies

- CloudKit, iCloud Drive, **NSUbiquitousKeyValueStore**, KeyChain
- **NSURLSession** for all networking
Move Your App Data Into the Cloud
Consistent user experience on different devices

iOS cloud technologies

- CloudKit, iCloud Drive, `NSUbiquitousKeyValueStore`, KeyChain
- `NSURLSession` for all networking

Flush data to cloud opportunistically

- Sync data before `applicationWillResignActive` events
- iOS will sync pending data when a user is not active
  - Background sync process for pending sync
CloudKit and iCloud Drive
CloudKit and iCloud Drive

CloudKit can store structured app and user data

• Syncing of data is taken care of automatically
• “LongLivedOperations” from the CKOperation Class will flush in background
CloudKit and iCloud Drive

CloudKit can store structured app and user data
- Syncing of data is taken care of automatically
- “LongLivedOperations” from the CKOperation Class will flush in background

Cloud Documents for all Managed Apple IDs
- Simple and easy for document based apps
CloudKit and iCloud Drive

CloudKit can store structured app and user data

- Syncing of data is taken care of automatically
- “LongLivedOperations” from the CKOperation Class will flush in background

Cloud Documents for all Managed Apple IDs

- Simple and easy for document based apps

Both of these techs are sync on demand

- Efficient fetch of data accessed by the account
CloudKit and iCloud Drive

CloudKit can store structured app and user data

- Syncing of data is taken care of automatically
- “LongLivedOperations“ from the CKOperation Class will flush in background

Cloud Documents for all Managed Apple IDs

- Simple and easy for document based apps

Both of these techs are sync on demand

- Efficient fetch of data accessed by the account

CloudKit is suitable for larger data sets
NSUbiquitousKeyValueStore
NSUbiquitousKeyValueStore

Requires an entitlement
- Drop in replacement for NSUserDefaults
- Easy to adopt for your app data
  - NSDictionary style data
  - Smaller payloads only please
NSUbiquitousKeyValueStore

Requires an entitlement
• Drop in replacement for NSUserDefaults
• Easy to adopt for your app data
  - NSDictionary style data
  - Smaller payloads only please

Data fetched as part of account sign-in
• Prepares your app to be managed via Application Management
• MDM deployments can provide initial application state
NSUbiquitousKeyValueStore
Example use case for Shared iPad

- Store your initial app setup data in NSUbiquitousKeyValueStore
- Apple apps use it for Stocks for ticker and locations for Weather
Keychain for Shared iPad
Keychain for Shared iPad

All Managed Apple IDs have a Keychain

- Restored and backed up each time an user signs in or signs out
Keychain for Shared iPad

All Managed Apple IDs have a Keychain

- Restored and backed up each time an user signs in or signs out

Same API and usage conventions
Keychain for Shared iPad

All Managed Apple IDs have a Keychain

• Restored and backed up each time an user signs in or signs out

Same API and usage conventions

Store user credentials necessary for external services in Keychain

• Make sure to mark the Keychain entries as “exportable”
Keychain for Shared iPad

All Managed Apple IDs have a Keychain
• Restored and backed up each time an user signs in or signs out

Same API and usage conventions

Store user credentials necessary for external services in Keychain
• Make sure to mark the Keychain entries as “exportable”

Don’t use the Keychain for bulk data
NSURLSession
NSURLConnection

Best option for existing cloud storage that isn’t iCloud-based

• Your app should be using NSURLSession for all networking operations
NSURLSession

Best option for existing cloud storage that isn’t iCloud-based
• Your app should be using NSURLSession for all networking operations

IPv6 cellular data control/accounting

robust data throttling

seamless network transport transition between Cellular/Wifi

Cisco fast lane support, etc…
Best option for existing cloud storage that isn’t iCloud-based

- Your app should be using NSURLSession for all networking operations

Optional background configuration

- Data sync when your app is not in the foreground
Shared iPad Best Practice
Shared iPad Best Practice

Cloud-based data is essential for your app

• Allows Managed Apple IDs to move between devices
Cloud-based data is essential for your app

• Allows Managed Apple IDs to move between devices

iOS cloud technologies provide complete solution
Shared iPad Best Practice

Cloud-based data is essential for your app

• Allows Managed Apple IDs to move between devices

iOS cloud technologies provide complete solution

iOS NSURLSession for all custom networking
Additional Considerations
Additional Considerations

Shared iPad is a powerful new feature in iOS 9.3

- Data separation for user accounts
- App shouldn’t store data for multiple users
  - Don’t co-mingling multiple user data: Rely on data separation
- Shared iPad will enforce quotas
  - Number of accounts cached on device
  - Amount of data stored per-user
  - Treat EDQUOT error same as ENOSPC
On-Demand Resources
On-Demand Resources

On-Demand Resources improve Shared iPad environments

- Using ODR avoids downloading assets for each user
  - Better performance, avoids quota use, less networking
  - App resources not duplicated across the device
  - Not purged when a user account is purged—better performance
On-Demand Resources

On-Demand Resources Guide

Setting Up On-Demand Resources

- **On-Demand Resources Essentials**
  - Enabling On-Demand Resources
  - Creating and Assigning Tags
  - Platform Sizes for On-Demand Resources

Managing On-Demand Resources

Hosting On-Demand Resources

Designing for On-Demand Resources

Revision History

<table>
<thead>
<tr>
<th>Type</th>
<th>Asset catalog</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data file</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Image</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>OpenGL shader</td>
<td>--</td>
<td>✓</td>
</tr>
<tr>
<td>SpriteKit particle</td>
<td>--</td>
<td>✓</td>
</tr>
<tr>
<td>SpriteKit scene</td>
<td>--</td>
<td>✓</td>
</tr>
<tr>
<td>SpriteKit texture atlas</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Apple TV Image Stack</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

A data file can contain any sort of data except for executable Swift, Objective-C, or C++ code. Files generated by scripting languages can be on-demand resources.

Benefits of On-Demand Resources

Some of the main ways apps can benefit from on-demand resources include:

- **Smaller app size.** The size of the app bundle downloaded by the user is smaller resulting in faster downloads and more storage room on the device.
- **Lazy loading of app resources.** The app has resources that are used only in certain states. The resources are requested when the app is likely to enter the appropriate state. For example, in a game with many levels, the user needs only the resources associated with the current and next levels.
- **Remote storage of rarely used resources.** The app has resources that are used infrequently. The resources are requested as they are needed. For example, an app tutorial is usually shown once after the app is opened for the first time, and may never be used again. The app requests the tutorial on first use.
Shared iPad Remote Notifications
Notifications work same as single user iPad
Notifications work same as single user iPad
Very much like being turned-on and turned-off
Shared iPad Remote Notifications

Notifications work same as single user iPad
Very much like being turned-on and turned-off
Remote notifications aren’t primed until your app runs at least once per device
• Just like a non-shared iPad
Shared iPad Remote Notifications

Notifications work same as single user iPad
Very much like being turned-on and turned-off
Remote notifications aren’t primed until your app runs at least once per device
• Just like a non-shared iPad
Signing out is like powering off a device
• Will not receive Remote notifications
Shared iPad Remote Notifications

Notifications work same as single user iPad
Very much like being turned-on and turned-off
Remote notifications aren’t primed until your app runs at least once per device
• Just like a non-shared iPad
Signing out is like powering off a device
• Will not receive Remote notifications
Review your usage of Remote notifications
Managed Apple IDs
Managed Apple IDs

Managed Apple IDs don’t have commerce features

• Make your app fully functional without in-app purchase
Managed Apple IDs

Managed Apple IDs don’t have commerce features

• Make your app fully functional without in-app purchase

Check for specific features/frameworks you need

• Managed Apple IDs will return proper errors for disabled features
  - StoreKit, for example, is disabled for Managed Apple IDs
Managed Apple IDs

Managed Apple IDs don’t have commerce features
• Make your app fully functional without in-app purchase
Check for specific features/frameworks you need
• Managed Apple IDs will return proper errors for disabled features
  - StoreKit, for example, is disabled for Managed Apple IDs

Follow existing app licensing strategies
• VPP and Device-based licensing is the deployment model for Shared iPad
Web Developers

If your content is web-based
• Consider moving it to a native app
• Customers overwhelmingly prefer this option
Web Developers

If your content is web-based
• Consider moving it to a native app
• Customers overwhelming prefer this option

If your website requires user authentication
• Consider Safari password autofill
• Shared iPad requires a Safari domain whitelist to allow password autofill for websites
• Update your documentation
  - This will allow Safari autofill
Cloud Is Not Just Shared iPad
Cloud Is Not Just Shared iPad

Cloud-based data is required for Shared iPad
Cloud Is Not Just Shared iPad

Cloud-based data is required for Shared iPad

Consider all your users

• Devices can be damaged, lost, stolen or upgraded - often are…
• Cloud based data makes it easy for all your customers
Cloud Is Not Just Shared iPad

Cloud-based data is required for Shared iPad

Consider all your users

• Devices can be damaged, lost, stolen or upgraded - often are…
• Cloud based data makes it easy for all your customers

Business use cases also favor the cloud

• More likely to transition devices and repurpose them as needs change and jobs change
Cloud Is Not Just Shared iPad

Cloud-based data is required for Shared iPad

Consider all your users

- Devices can be damaged, lost, stolen or upgraded - often are…
- Cloud based data makes it easy for all your customers

Business use cases also favor the cloud

- More likely to transition devices and repurpose them as needs change and jobs change

Cloud data storage is the long-term trend
Testing Your App for Shared iPad
Testing Your App for Shared iPad

Three methods for testing

- Two devices: Verify data propagates between devices
- Same device: Add/remove your app
- Extreme: Erase the device
  - Configure user account: See if all your app data comes back
Testing Your App for Shared iPad

Three methods for testing

• Two devices: Verify data propagates between devices
• Same device: Add/remove your app
• Extreme: Erase the device
  - Configure user account: See if all your app data comes back

Verify your app’s functionality

• No data loss, offline use, initial app configuration only presented once
Testing Your App for Shared iPad

Three methods for testing

• Two devices: Verify data propagates between devices
• Same device: Add/remove your app
• Extreme: Erase the device
  - Configure user account: See if all your app data comes back

Verify your app’s functionality

• No data loss, offline use, initial app configuration only presented once

Make sure your app doesn’t block sign-out

• Don’t leak UIKitBackgroundTask
Opportunities to Go Beyond the Cloud
Classroom App

Meet your new teaching assistant.

Classroom is a powerful new iPad app that helps teachers guide learning, share work, and manage student devices. Talk with your technology manager about setting it up for your school.

Watch the Classroom app Guided Tour ©
Universal Links
Universal Links

Universal Links introduced with iOS 9
Universal Links

Universal Links introduced with iOS 9
Powerful feature that allows shared links
Universal Links

Universal Links introduced with iOS 9

Powerful feature that allows shared links

iOS can search inside your app

• Required for the share sheet
• Quickly navigate to the same place on two different devices
Universal Links

Universal Links introduced with iOS 9

Powerful feature that allows shared links

iOS can search inside your app

• Required for the share sheet
• Quickly navigate to the same place on two different devices

New for this year is Classroom app

• Teachers can integrate into class curriculum
What’s New in iOS 9.3.2

Automatic assessment configuration

Continues to work same way on supervised devices

New entitlement

API then disables five features while app is running

• Auto correction, Define, keyboard shortcuts, predictive keyboard, spell check

Safe escape behavior on unmanaged devices
Managed App Configuration
Managed App Configuration

Introduced with iOS 7
Managed App Configuration

Introduced with iOS 7

Allows default app configuration on an institutional scale

• Additional key in NSUserDefaults
• Key presence leads to an app specific dictionary
Managed App Configuration

Introduced with iOS 7

Allows default app configuration on a institutional scale

- Additional key in NSUserDefaults
- Key presence leads to a app specific dictionary

AppConfiguration community is new this year

- MDM developers standardizing payloads
- http://appconfig.org
- One app - multi-MDM support
Managed Application App Adopters

iPhone

iPad
Managed Application App Adopters
Best Apps Leverage iOS
Best Apps Leverage iOS
More Information

https://developer.apple.com/wwdc16/304
## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>What’s New in Apple Device Management</td>
<td>Nob Hill</td>
<td>Wednesday 11:00AM</td>
</tr>
<tr>
<td>Improving Existing Apps with Modern Best Practices</td>
<td>Presidio</td>
<td>Wednesday 3:00PM</td>
</tr>
<tr>
<td>Optimizing On-Demand Resources</td>
<td>Mission</td>
<td>Thursday 10:00AM</td>
</tr>
<tr>
<td>Extending Your Apps with SiriKit</td>
<td>Nob Hill</td>
<td>Thursday 1:40PM</td>
</tr>
<tr>
<td>What’s New with CloudKit</td>
<td>Presidio</td>
<td>Thursday 3:00PM</td>
</tr>
<tr>
<td>CloudKit Best Practices</td>
<td>Pacific Heights</td>
<td>Friday 9:00AM</td>
</tr>
<tr>
<td>Labs</td>
<td>Frameworks Lab</td>
<td>Time</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>----------------</td>
<td>---------------</td>
</tr>
<tr>
<td>NSUserActivity, Universal Links and Handoff Lab</td>
<td>Lab A</td>
<td>Tuesday 10:00AM</td>
</tr>
<tr>
<td>Accessibility and Speech Lab</td>
<td>Lab C</td>
<td>Wednesday 2:30PM</td>
</tr>
</tbody>
</table>