Wallet—The Home for Apple Pay and More

Session 701

Danvin Ruangchan iOS Apps & Frameworks
What We’ve Been Up To

A Review of Wallet Features

What is a Pass?

PassKit

Apple Pay

And More…
What Have We Been Up to...
What Have We Been Up to...
Wallet

How did we get here?
Header Fields
Secondary and Auxiliary Fields
CONFIRMATION
A1B2C3

CABIN
UNITED ECONOMY (XN)

FLIGHT DATE
TUE. JUN. 9, 2015

DEPARTS
5:15 PM
What Makes a Pass, a Pass
What Makes a Pass, a Pass

pass.json
What Makes a Pass, a Pass
What Makes a Pass

Pass.pkpass
- pass.json
- fr.lproj
  - pass.strings
- de.lproj
  - pass.strings
- icon.png
- icon@2x.png
- logo.png
- logo@2x.png
- strip.png
- strip@2x.png
- manifest.json
- signature
What Makes a Pass, Your Pass

Team Identifier

- 10-character string found on the developer portal
- Used in entitling apps to access passes through PassKit

"teamIdentifier" : "DK9N2M2GK6"
What Makes a Pass, Your Pass

Pass Type Identifier

- Defines a class or category of passes
- Registered on the developer portal
- Used for grouping

"passTypeIdentifier" : "pass.com.mycompany.storecard"
What Makes a Pass, Your Pass

Serial Number
• Unique for every pass
• Chosen by you and opaque to Wallet
• Combined with Pass Type Identifier to determine global uniqueness

"serialNumber" : "B5BD0271-B90B-400D-8344-36A789714CC8"
Relevancy

Date

Location

Beacons
"locations" : [ 
{
   "longitude" : -118.2400,
   "latitude" : 34.0736,
   "relevantText" : "My Store - Los Angeles"
},
{
   "longitude" : -122.03118,
   "latitude" : 37.33182,
   "relevantText" : "My Store - Bay Area"
}
],

"relevantDate" : "2015-06-09T13:00-18:00"
Relevancy

"beacons": [
    {
        "proximityUUID":"F8F589E9-C07E-58B0-AEAB-A36BE4D48FAC",
        "relevantText":"You're near my store",
        "name": "My Store"
    }
],
Keep Passes Up to Date

Apple Push Service

passTypeID

Get Pass Updates

webService

Updated Passes

.pkplass

Your Server
PKPassLibrary

- Query user’s passes
- Access gated by entitlements
PKPassLibrary

class func isPassLibraryAvailable() -> Bool

func passes() -> [PKPass]

func passWithNamePassTypeIdentifier(identifier: String,
    serialNumber: String) -> PKPass?
PKPassLibrary

class func isPassLibraryAvailable() -> Bool

func passes() -> [PKPass]

func passWithPassTypeIdentifier(identifier: String, serialNumber: String) -> PKPass?
class func isPassLibraryAvailable() -> Bool

func passes() -> [PKPass]

func passWithPassTypeIdentifier(identifier: String, serialNumber: String) -> PKPass?
class func isPassLibraryAvailable() -> Bool

func passes() -> [PKPass]

func passWithPassTypeIdentifier(identifier: String, serialNumber: String) -> PKPass?
func passesOfType(passType: PKPassType) -> [PKPass]
func passesOfType(passType: PKPassType) -> [PKPass]

passLibrary.pusesOfType(.Barcode)
passLibrary.pusesOfType(.Payment)
passLibrary.pusesOfType(.Any)
Distributing Passes

Web

Email

App
Distributing Passes
func addPasses(passes: [PKPass],
withCompletionHandler completion: ((PKPassLibraryAddPassesStatus) -> Void)?)
PKPassLibrary

func addPasses(passes: [PKPass],
withCompletionHandler completion: ((PKPassLibraryAddPassesStatus) -> Void)?)
PKAddPassesViewController
PKAddPassesViewController

class func canAddPasses() -> Bool

init!(pass pass: PKPass)

init!(passes passes: [PKPass])
PKAddPassesViewController

class func canAddPasses() -> Bool

init!(pass pass: PKPass)

init!(passes passes: [PKPass])
PKAddPassesViewController

class func canAddPasses() -> Bool

init!(pass pass: PKPass)

init!(passes passes: [PKPass])
PKAddPassesViewController

class func canAddPasses() -> Bool

init!(pass pass: PKPass)

init!(passes passes: [PKPass])
Engaging Passes

- Relevancy
- Update
- Images
- Formatting
Engaging Passes

- Relevancy
- Update
- Images
- Formatting
Engaging Passes

• Relevancy
• Update
• Images
• Formatting
Engaging Passes

- Relevancy
- Update
- Images
- Formatting
Wallet

Where do we go from here?
Passes That Work with Apple Pay
Passes That Work with Apple Pay

- Contactless
- Value Added Services Protocol
- Working with terminal manufacturers
"nfc" : {
    "message" : "4444678966661234",
    "encryptionPublicKey" : "MDkwEwYHKoZIzj0CAQYIzj0DAQcDIgACigjq+QYC17m3i9uO8uKc8mLpaS1UJ0EaCFvMedkXsuA="
},
Apple Pay
Why Encourage Apple Pay Cards?

- Seamless payment experience
- Higher conversion rates
PassKit + Apple Pay

Adding Payment Passes

• Link to Wallet
• Within an app
PKPassLibrary
PKPassLibrary

func openPaymentSetup()
PKAddPaymentPassViewController

class func canAddPaymentPass() -> Bool

init?(requestConfiguration configuration: PKAddPaymentPassRequestConfiguration,
delegate delegate: PKAddPaymentPassViewControllerDelegate?)
PKAddPaymentPassViewController

class func canAddPaymentPass() -> Bool

init?(requestConfiguration configuration: PKAddPaymentPassRequestConfiguration,
delegate delegate: PKAddPaymentPassViewControllerDelegate?)
PKAddPaymentPassViewControllerDelegate
func addPaymentPassViewController(controller: PKAddPaymentPassViewController, generateRequestWithCertificateChain certificates: [NSData], nonce nonce: NSData, nonceSignature nonceSignature: NSData, completionHandler handler: (PKAddPaymentPassRequest) -> Void)
Payment Pass Verification

Card Verification
Choose how to verify your card for Apple Pay.

- Capital One App
- Capital One Wallet

Call Capital One 866-381-0451
Payment Pass Verification

Card Verification
Choose how to verify your card for Apple Pay.

- Capital One App
- Capital One Wallet
- Call Capital One 866-381-0451
func isPaymentPassActivationAvailable() -> Bool

func activatePaymentPass(paymentPass: PKPaymentPass,
    withActivationData activationData: NSData,
    completion: ((Bool, NSError) -> Void)?)
func isPaymentPassActivationAvailable() -> Bool

func activatePaymentPass(paymenPass: PKPaymentPass, 
withActivationData activationData: NSData, 
completion: ((Bool, NSError) -> Void)?)
func isPaymentPassActivationAvailable() -> Bool

func activatePaymentPass(paymnetPass: PKPaymentPass,
withActivationData activationData: NSData,
completion: ((Bool, NSError) -> Void)?)
Contactless Interface Suppression

- Temporarily disables payment UI
- Requires user’s permission
class func requestAutomaticPassPresentationSuppressionWithResponseHandler(
  responseHandler: (PKAutomaticPassPresentationSuppressionResult) ->
  PKSuppressionRequestToken)

class func endAutomaticPassPresentationSuppressionWithRequestToken(
  requestToken: PKSuppressionRequestToken)

class func isSuppressingAutomaticPassPresentation() -> Bool
class func requestAutomaticPassPresentationSuppressionWithResponseHandler(
    responseHandler: (PKAutomaticPassPresentationSuppressionResult) ->
    PKSuppressionRequestToken)

class func endAutomaticPassPresentationSuppressionWithRequestToken(
    requestToken: PKSuppressionRequestToken)

class func isSuppressingAutomaticPassPresentation() -> Bool
class func requestAutomaticPassPresentationSuppressionWithResponseHandler(
    responseHandler: (PKAutomaticPassPresentationSuppressionSuppressionResult) ->
    PKSuppressionRequestToken)

class func endAutomaticPassPresentationSuppressionWithRequestToken(
    requestToken: PKSuppressionRequestToken)

class func isSuppressingAutomaticPassPresentation() -> Bool
class func requestAutomaticPassPresentationSuppressionWithResponseHandler (responseHandler: (PKAutomaticPassPresentationSuppressionResult) -> PKSuppressionRequestToken)

class func endAutomaticPassPresentationSuppressionWithRequestToken (requestToken: PKSuppressionRequestToken)

class func isSuppressingAutomaticPassPresentation() -> Bool
PassKit + Apple Watch
PassKit + Apple Watch

- Add
- Remove
- Query
PassKit + Apple Watch
PassKit + Apple Watch

- Passes look different
- Follow H1 guidelines
1D Barcodes
1D Barcodes

New barcode format:

PKBarcodeFormatCode128
1D Barcodes

"barcode": {
    "message": "123456789",
    "format": "PKBarcodeFormatPDF417",
    "messageEncoding": "iso-8859-1"
},
"barcodes": [
    {
        "message": "Hello World Code 128",
        "format": "PKBarcodeFormatCode128",
        "messageEncoding": "iso-8859-1"
    },
    {
        "message": "Hello World QR Code",
        "format": "PKBarcodeFormatPDF417",
        "messageEncoding": "iso-8859-1"
    }
]
1D Barcodes

• Verify your system supports 1D barcodes
  - Optical scanners
• Fall back to 2D barcodes on iOS 8 and below
Remember…

Solving the overstuffed wallet problem
• Passes allow user to take action in the real world
• Easy to integrate
• Keep passes engaging
Integrate Apple Pay into your app
Use Wallet and Apple Pay
More Information

Documentation and Videos
Passbook for Developers

Apple Pay for Developers

Introducing Passbook Part 1 & 2 - WWDC 2012

Integrating Passbook into Your Ecosystem - WWDC 2013
More Information

Technical Support
Apple Developer Forums
http://developer.apple.com/forums

Developer Technical Support
http://developer.apple.com/support/technical

General Inquiries
Paul Danbold, Core OS Evangelist
danbold@apple.com
## Related Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introducing WatchKit for watchOS 2</td>
<td>Presidio</td>
<td>Tuesday 10:00AM</td>
</tr>
<tr>
<td>Apple Pay Within Apps</td>
<td>Mission</td>
<td>Tuesday 11:00AM</td>
</tr>
<tr>
<td>Labs</td>
<td>Frameworks Lab C</td>
<td>Time</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Wallet and Apple Pay Lab</td>
<td></td>
<td>Tuesday 1:30PM</td>
</tr>
<tr>
<td>Wallet and Apple Pay Lab</td>
<td></td>
<td>Wednesday 1:30PM</td>
</tr>
</tbody>
</table>