#WWDC19

Creating Independent Watch Apps

Neil Desai, watchOS Frameworks Engineer

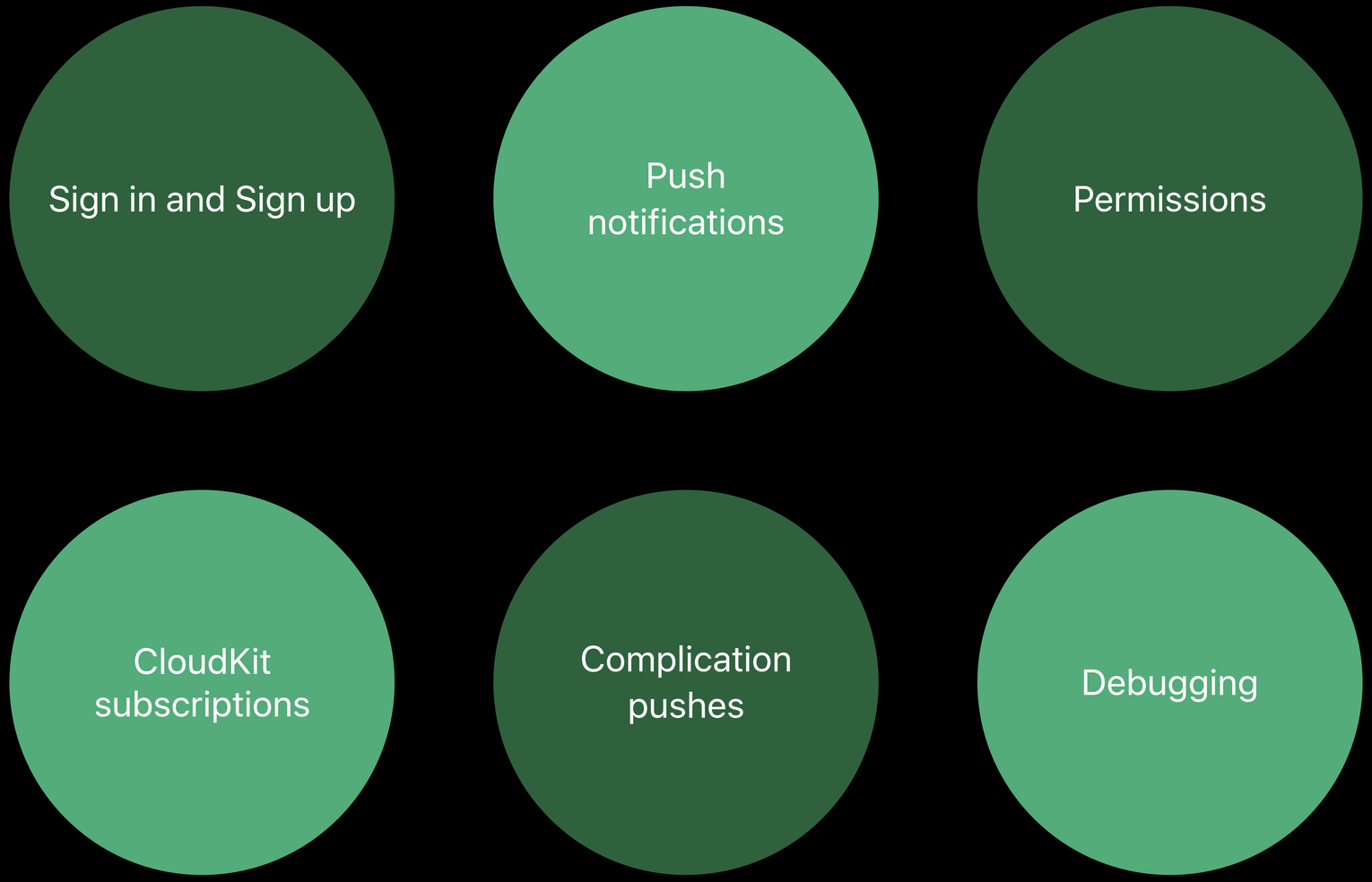

9:41
FaceTime
Monday
3
Calendar
Photos
Camera
Mail
Clock
Maps
280
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
tv
TV
Health
Home
Wallet
Settings


9:41
FaceTime
Monday
3
Calendar
Photos
Camera
Mail
Clock
Maps
280
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
tv
TV
Health
Home
Wallet
Settings


Mon
3

Independent Watch Apps


Outcast
10:09
The Screaming Spectr...
Do By Friday
15
45
33:41
-57:21
San Jose
10:09
72°
12%
I'm only bringing out the sun so I can blind you with it.
10AM
1
4
7PM
72
76
75
70
INR → USD
10:09
INR
10
0.14
USD
Reset


Sign in and Sign up
Push notifications
Permissions
CloudKit subscriptions
Complication pushes
Debugging


Mon
3


Mon
3


Mon
3


Mon
3

Installation

Installation

iOS app no longer includes watch app

Installation

iOS app no longer includes watch app

Each device downloads its own app

Installation

iOS app no longer includes watch app

Each device downloads its own app

Asset and variant thinning

Installation

iOS app no longer includes watch app

Each device downloads its own app

Asset and variant thinning

Downloads are smaller


Ask to buy
Purchase history
Updates
Pre-orders
Siri search
Purchase
Search in-app
Browse


App Store
10:09
Search
Listen on the Go
Search
10:09
Dictation
Scribble
TRENDING
exercise
MLB At Bat
10:09
GET
In-App Purchases
MLB At Bat
The #1 App For Live Baseball
4.6, 36.2 Ratings
4+
Age

Dependent Apps

Dependent Apps

Watch app download will install iPhone app to iPhone

Dependent Apps

Watch app download will install iPhone app to iPhone

watchOS app launch is blocked until iPhone app is installed

Independent Apps

Independent Apps

Watch app is installed independently

Independent Apps

Watch app is installed independently

Can now uninstall the iPhone app and Watch app can remain

Independent Apps

Watch app is installed independently

Can now uninstall the iPhone app and Watch app can remain

watchOS app with iOS app is backwards compatible

Independent Apps

Watch app is installed independently

Can now uninstall the iPhone app and Watch app can remain

watchOS app with iOS app is backwards compatible

Watch-only app is watchOS 6 or later

Enterprise Distribution

Enterprise Distribution

Xcode support

Enterprise Distribution

Xcode support

Variants

Enterprise Distribution

Xcode support

Variants

`platform-identifier` key

Make Your Watch Apps Independent

Demo

Apps in Xcode

Apps in Xcode

Migrate your existing Xcode project

Apps in Xcode

Migrate your existing Xcode project

Watch-only app

Apps in Xcode

Migrate your existing Xcode project

Watch-only app

Simulator experience

Debugging

Debugging

Simulator debugging is up to 10x faster

Debugging

Simulator debugging is up to 10x faster

Device debugging is up to 2x faster

Debugging

Simulator debugging is up to 10x faster

Device debugging is up to 2x faster

Proxied through iPhone



Sign In and Sign Up

Sign In and Sign Up

Sign up

Sign In and Sign Up

Sign up

Terms & Conditions, use `WKAlertAction` API

Sign In and Sign Up

Sign up

Terms & Conditions, use `WKAlertAction` API

Sign In with Apple

Sign In and Sign Up

Sign up

Terms & Conditions, use `WKAlertAction` API

Sign In with Apple

Your own password sign in

Sign In with Apple

Sign In with Apple

NEW

Simple and secure

Sign In with Apple

NEW

Simple and secure

No forms

Sign In with Apple

NEW

Simple and secure

No forms

No new password

Sign In with Apple

NEW

Simple and secure

No forms

No new password

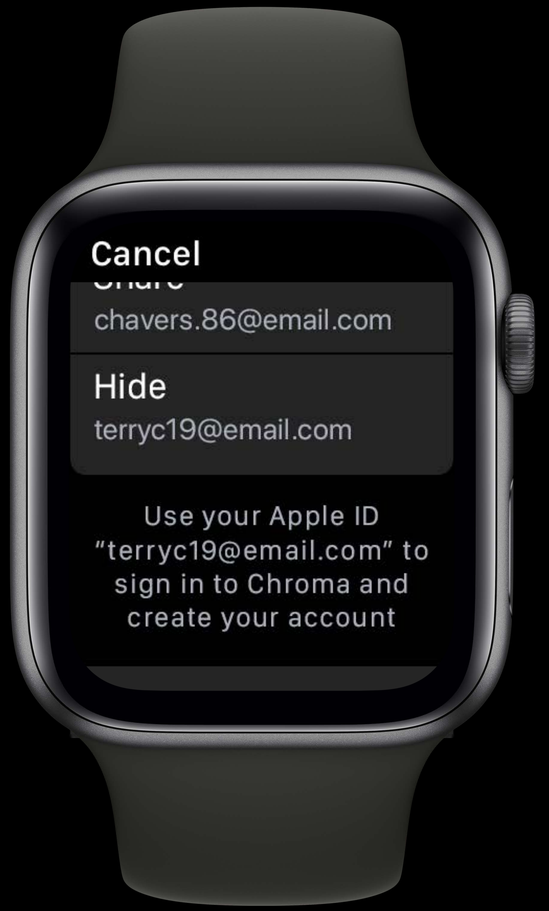
Sign In with Apple

NEW

Simple and secure

No forms

No new password



Sign In with Apple

NEW

Simple and secure

No forms

No new password

Sign In with Apple

NEW

Simple and secure

No forms

No new password

Two-factor authentication for every account

Sign In with Apple

NEW

Simple and secure

No forms

No new password

Two-factor authentication for every account

No email verification

Sign In with Apple

NEW

Simple and secure

No forms

No new password

Two-factor authentication for every account

No email verification

Sign-in across devices

Sign In with Apple

Sign In with Apple

Use AuthenticationServices.framework

Sign In with Apple

Use AuthenticationServices.framework

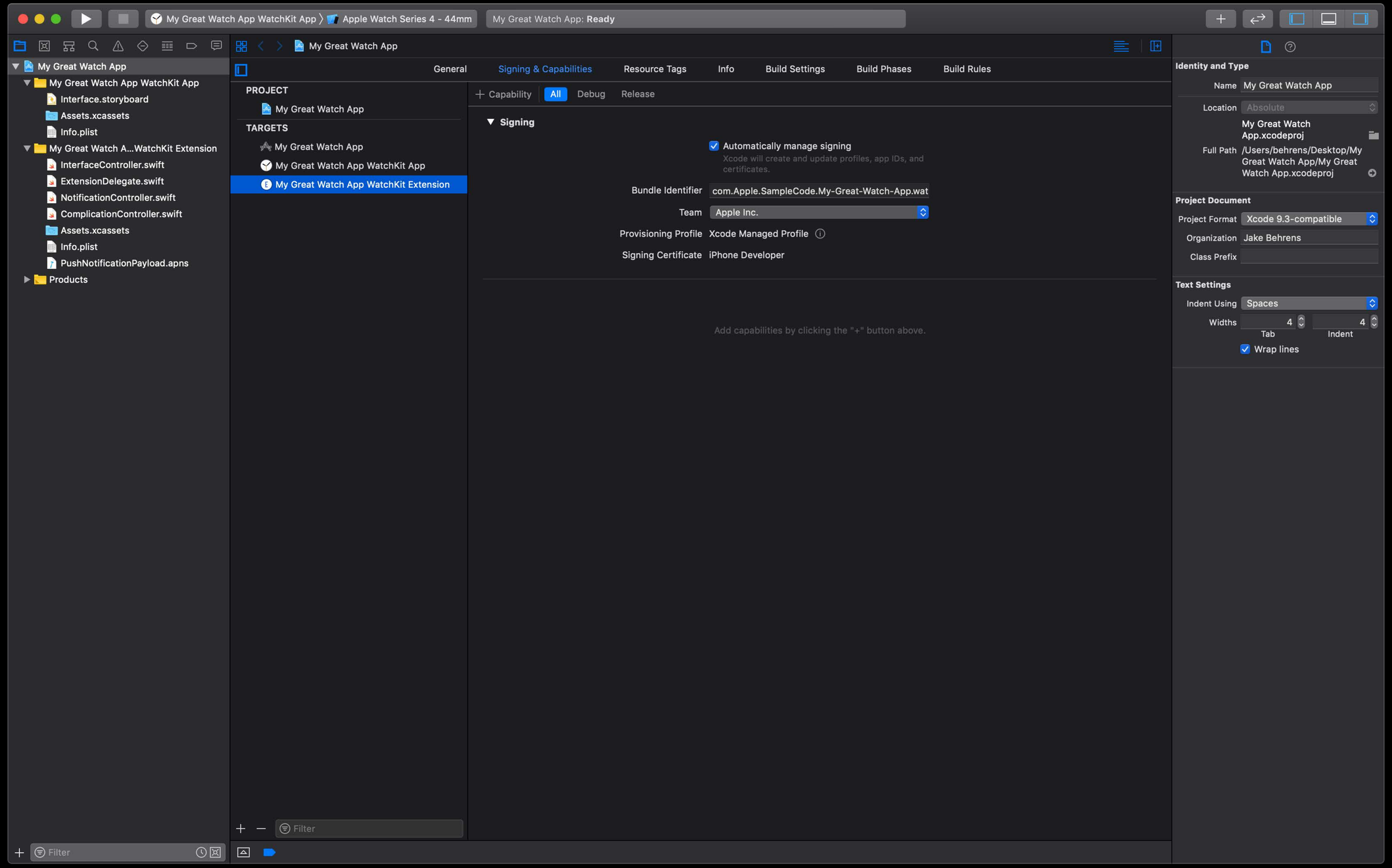
```
WKInterfaceAuthorizationAppleIDButton
```

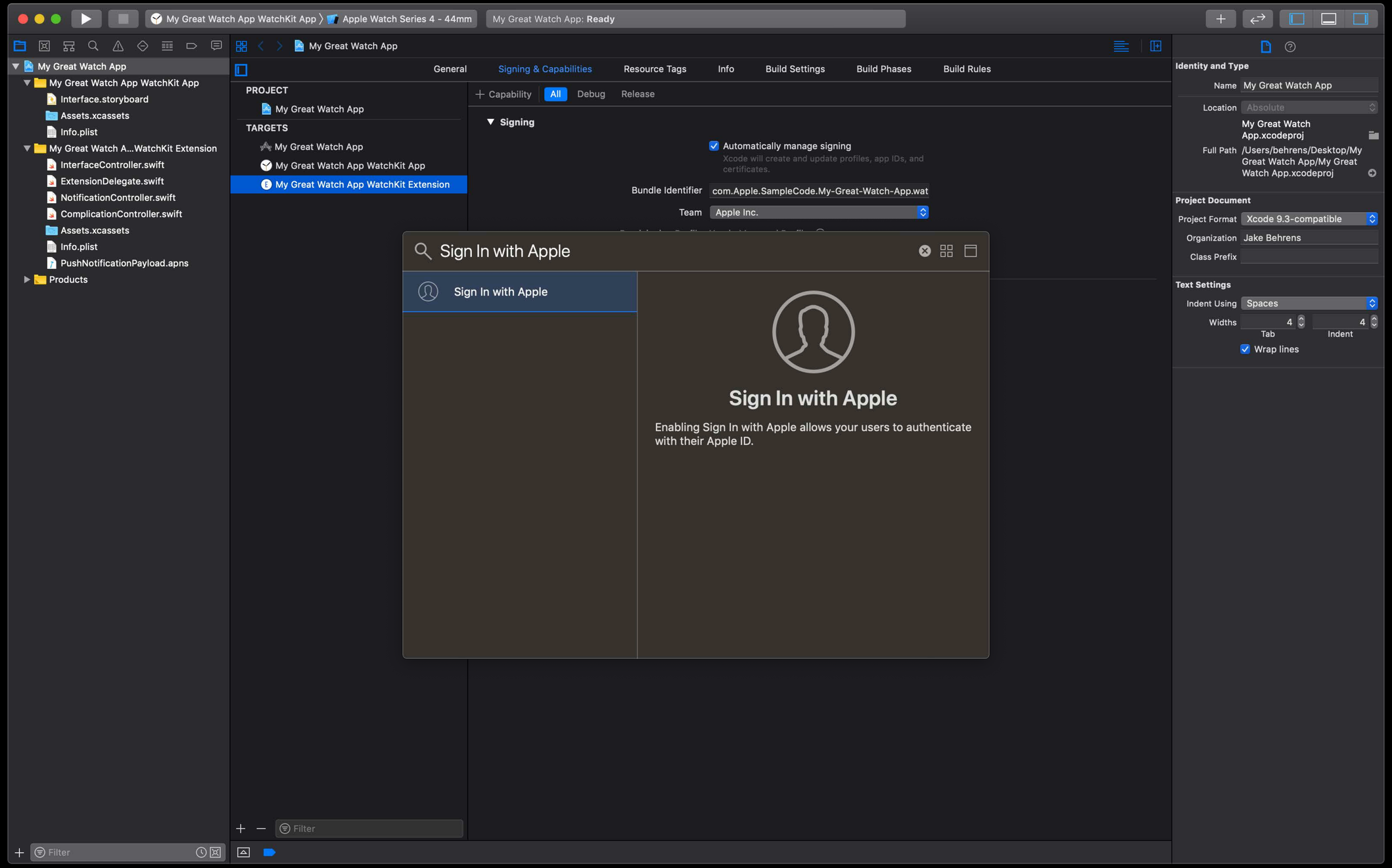
Sign In with Apple

Use AuthenticationServices.framework

`WKInterfaceAuthorizationAppleIDButton`

Introducing Sign In with Apple	Wednesday, 9:00
What’s New in Authentication	Thursday, 11:00


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
Filter
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Filter
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Signing Certificate
iPhone Developer
Add capabilities by clicking the "+" button above.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
4
Tab
Indent
Wrap lines


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Sign In with Apple
Sign In with Apple
Sign In with Apple
Enabling Sign In with Apple allows your users to authenticate with their Apple ID.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
Tab
4
Indent
Wrap lines
Filter
Filter

TextField

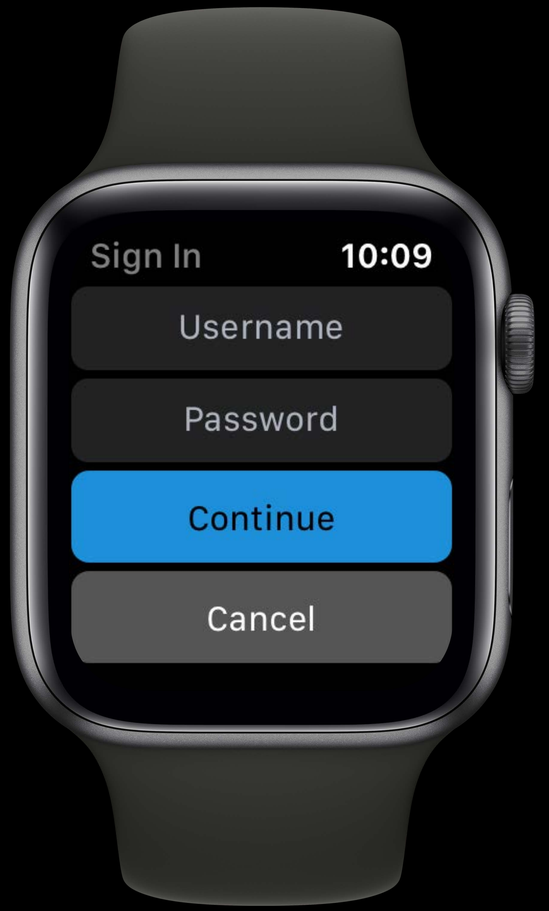
TextField

Embed TextField

TextField

Embed TextField

Use placeholder to instruct user what to input

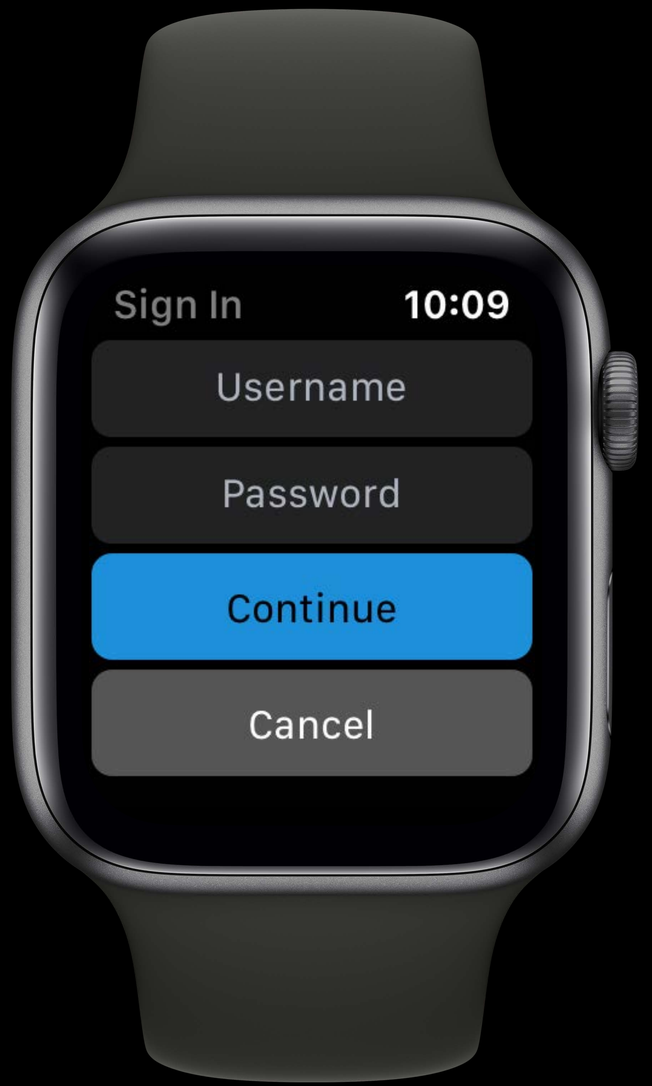


TextField

Embed TextField

Use placeholder to instruct user what to input

Set `textContentType`


Sign In
10:09
Username
Password
Continue
Cancel


Cancel


Cancel


9:41 AM Mon Jun 3
100%
APPLE WATCH KEYBOARD
Keyboard Input
Enter text on your Apple Watch using your iOS keyboard. Press to type.
FaceTime
Calendar
Monday
3
Clock
Home
Photos
Camera
Reminders
Notes
Voice Memos
Contacts
Maps
Find My
Books
Podcasts
TV
News
Stocks
Settings
9:41
APPLE WATCH KEYBOARD
Keyboard Input
Enter text on your Apple Watch using your iOS keyboard. Press to type.
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Home
Wallet
Settings
Cancel


9:41 AM Mon Jun 3
100%
APPLE WATCH KEYBOARD
Keyboard Input
Enter text on your Apple Watch using your iOS keyboard. Press to type.
FaceTime
Calendar
Monday
3
Clock
Home
Photos
Camera
Reminders
Notes
Voice Memos
Contacts
Maps
Find My
Books
Podcasts
TV
News
Stocks
Settings
9:41
APPLE WATCH KEYBOARD
Keyboard Input
Enter text on your Apple Watch using your iOS keyboard. Press to type.
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Home
Wallet
Settings
Done
Apple Watch Keyboard
Finish entering text on your iOS device.


APPLE WATCH KEYBOARD
Passwords
q w e r t y u i o p
a s d f g h j k l
z x c v b n m
123
space
return

Password AutoFill Suggestions on iOS

Password AutoFill Suggestions on iOS

Set the correct `textContentType`

Password AutoFill Suggestions on iOS

Set the correct `textContentType`

Associated domains

Password AutoFill Suggestions on iOS

Set the correct `textContentType`

Associated domains

User selection autofills both username and password


One time code 10:09
One time code
Sign in


Cancel
953121
From Messages

TextField

TextField

TextField

TextField

TextField

Associated domains

TextField

TextField

Associated domains

Continuity keyboard

TextField

TextField

Associated domains

Continuity keyboard

One time code

Privacy Management

Privacy Management

Calendar, contacts, motion

Privacy Management

Calendar, contacts, motion

Location

Privacy Management

Calendar, contacts, motion

Location

Health

Privacy Management

Calendar, contacts, motion

Location

Health

Privacy Management

Calendar, contacts, motion

Location

Health

Privacy Management

Calendar, contacts, motion

Location

Health





Push Notifications

Push Notifications

NEW

Watch as a push target

Push Notifications

NEW

Watch as a push target

User-visible notifications

Push Notifications

NEW

Watch as a push target

User-visible notifications

Background notifications

Notifications

APNs request header

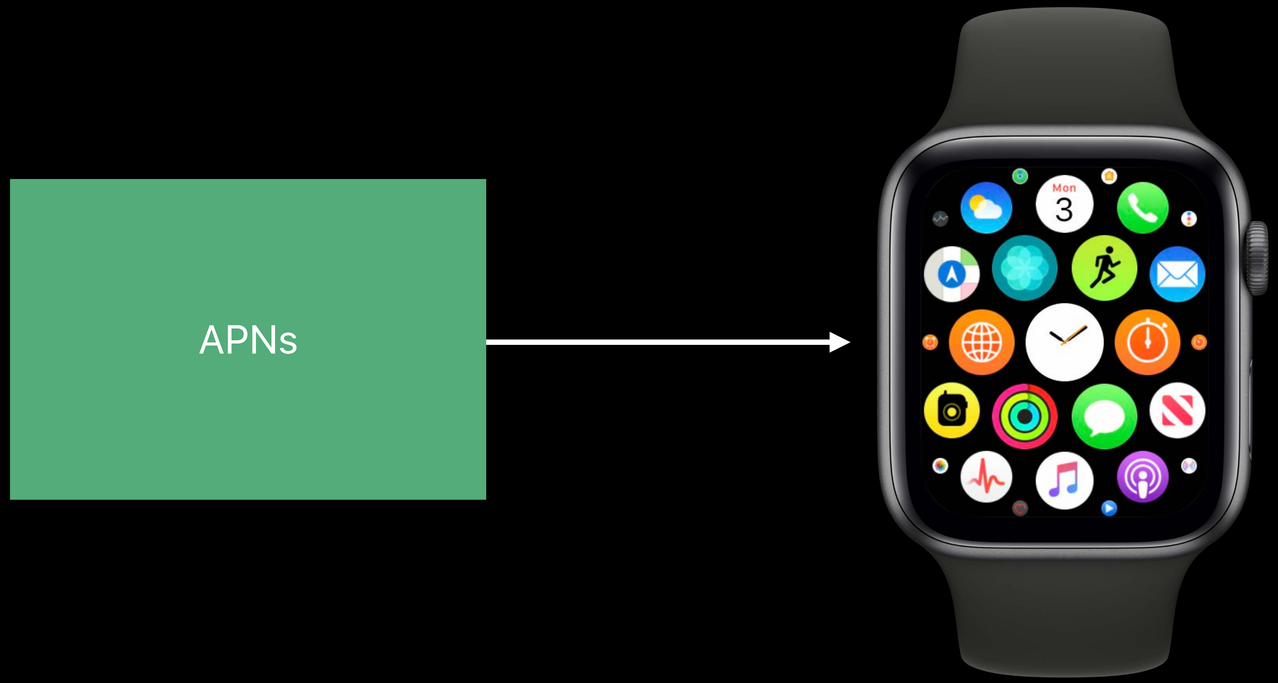
Payload

APNs

Notifications

APNs

Notifications



APNs Request Header

APNs Request Header

NEW

`apns-push-type`

APNs Request Header

NEW

`apns-push-type`

`alert`

APNs Request Header

NEW

`apns-push-type`

`alert`

`background`

Push Notifications

Registration and delivery

Push Notifications

Registration and delivery

WatchKit for registration

Push Notifications

Registration and delivery

WatchKit for registration

UserNotifications

Push Notifications

Registration and delivery

WatchKit for registration

UserNotifications

Background notifications delivered on WKExtensionDelegate

Push Notifications

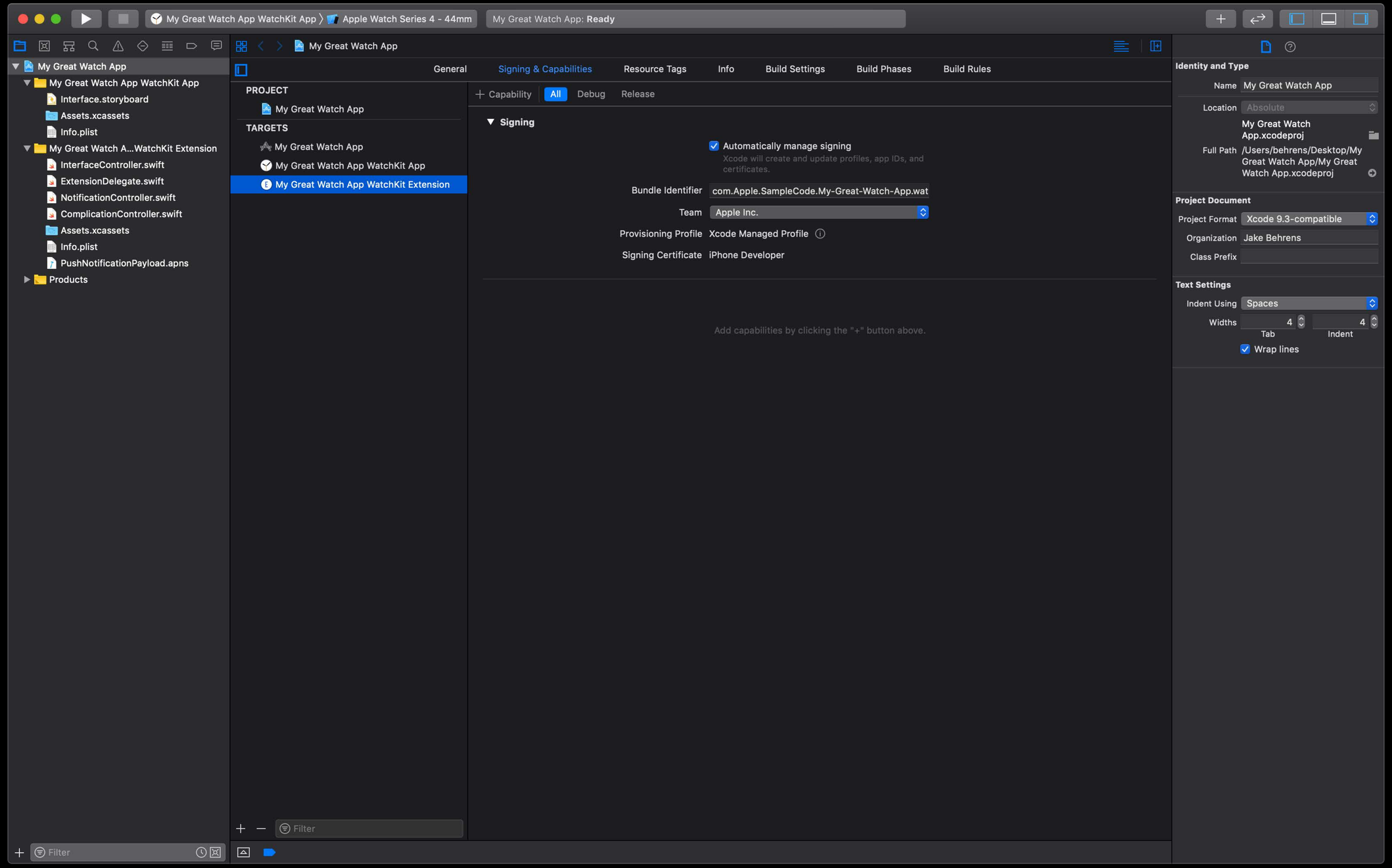
Registration and delivery

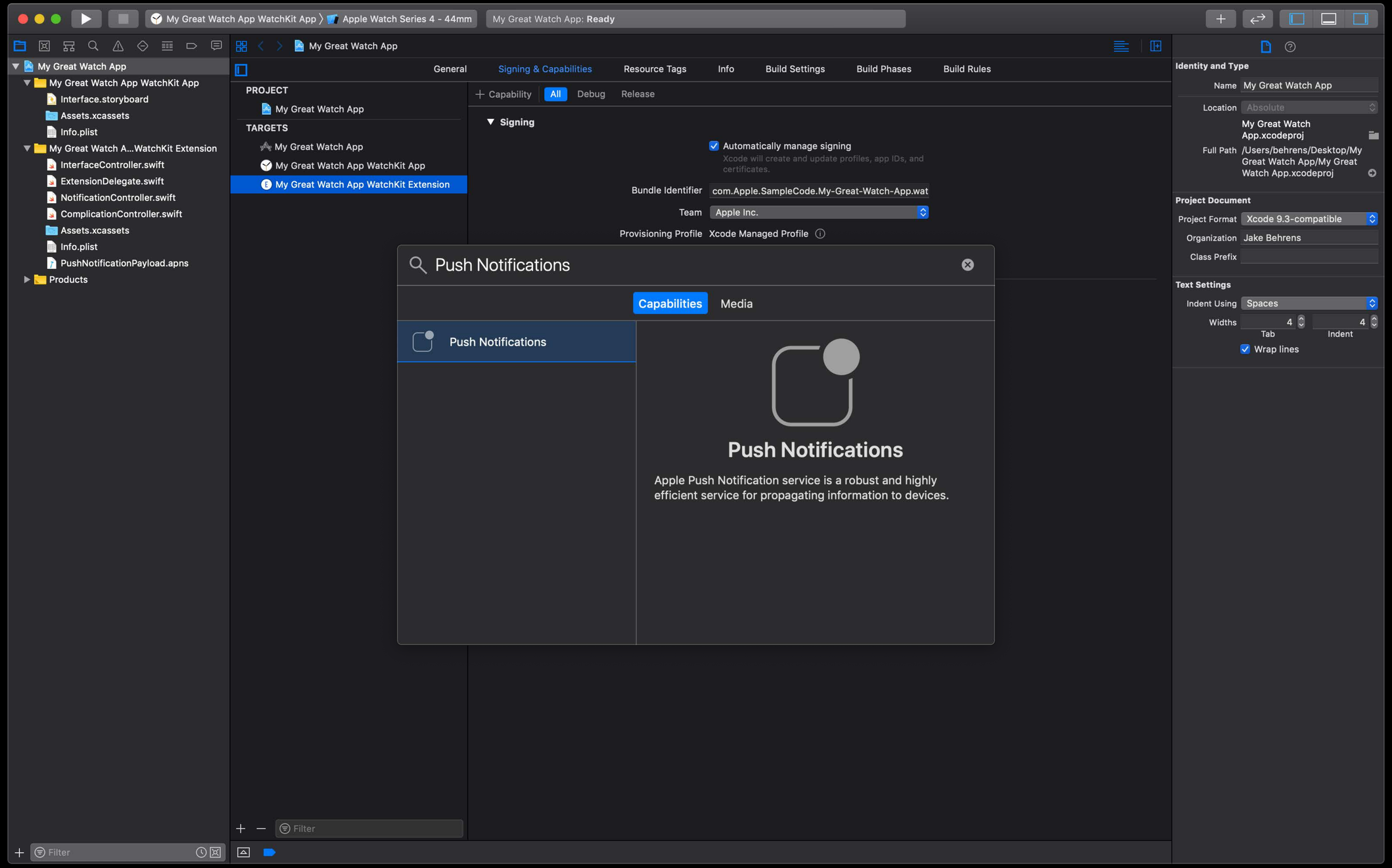
WatchKit for registration

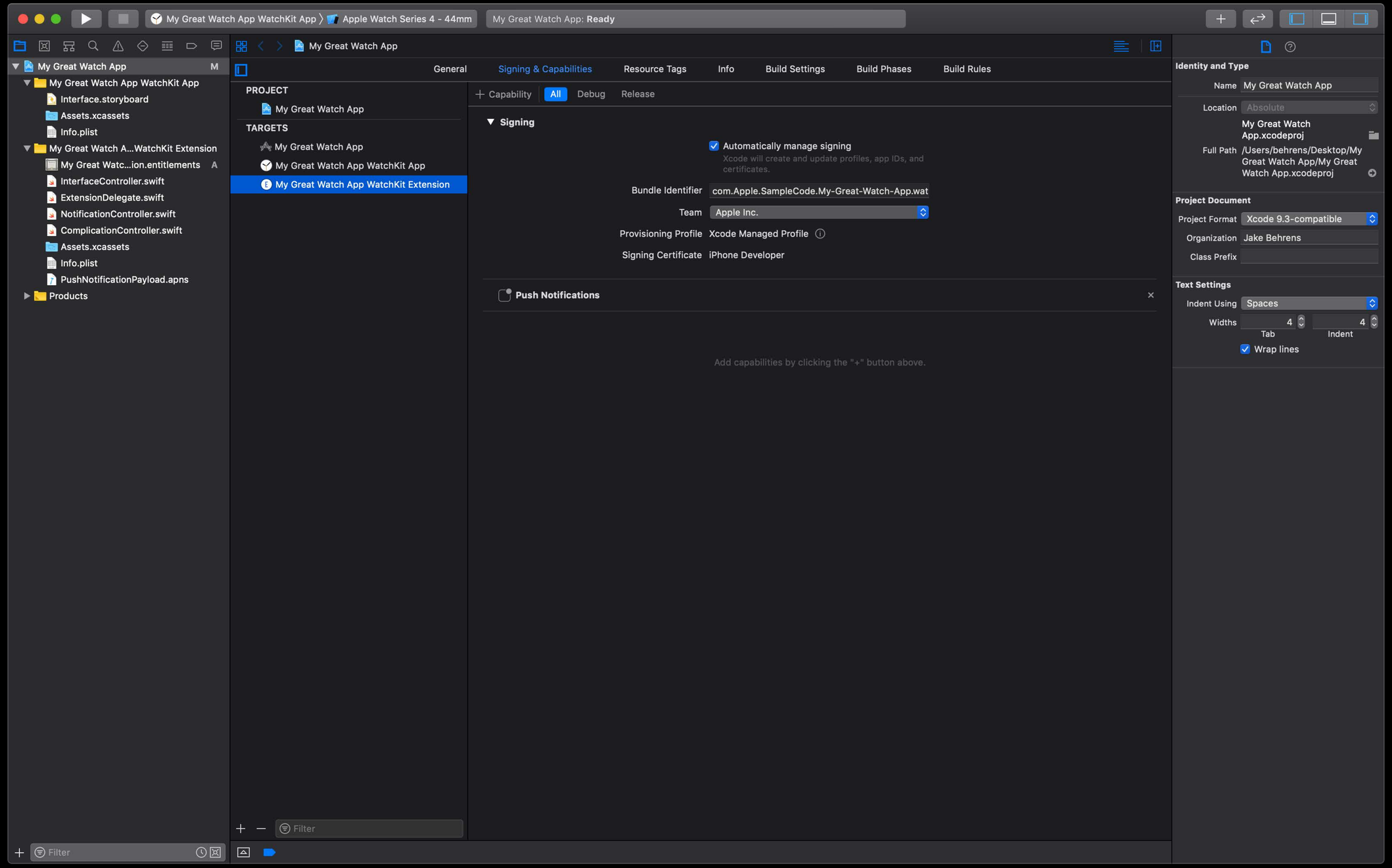
UserNotifications

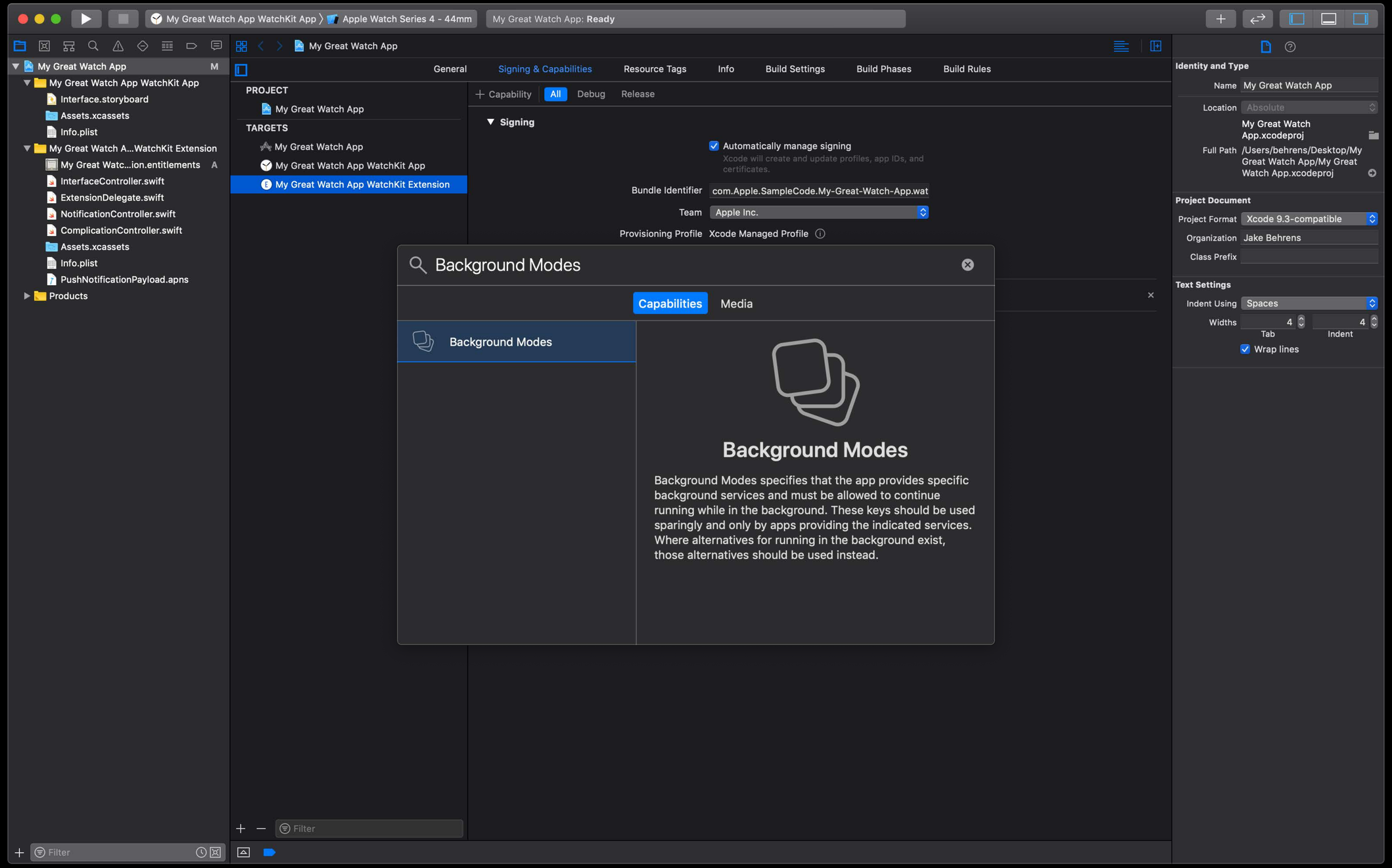
Background notifications delivered on WKExtensionDelegate

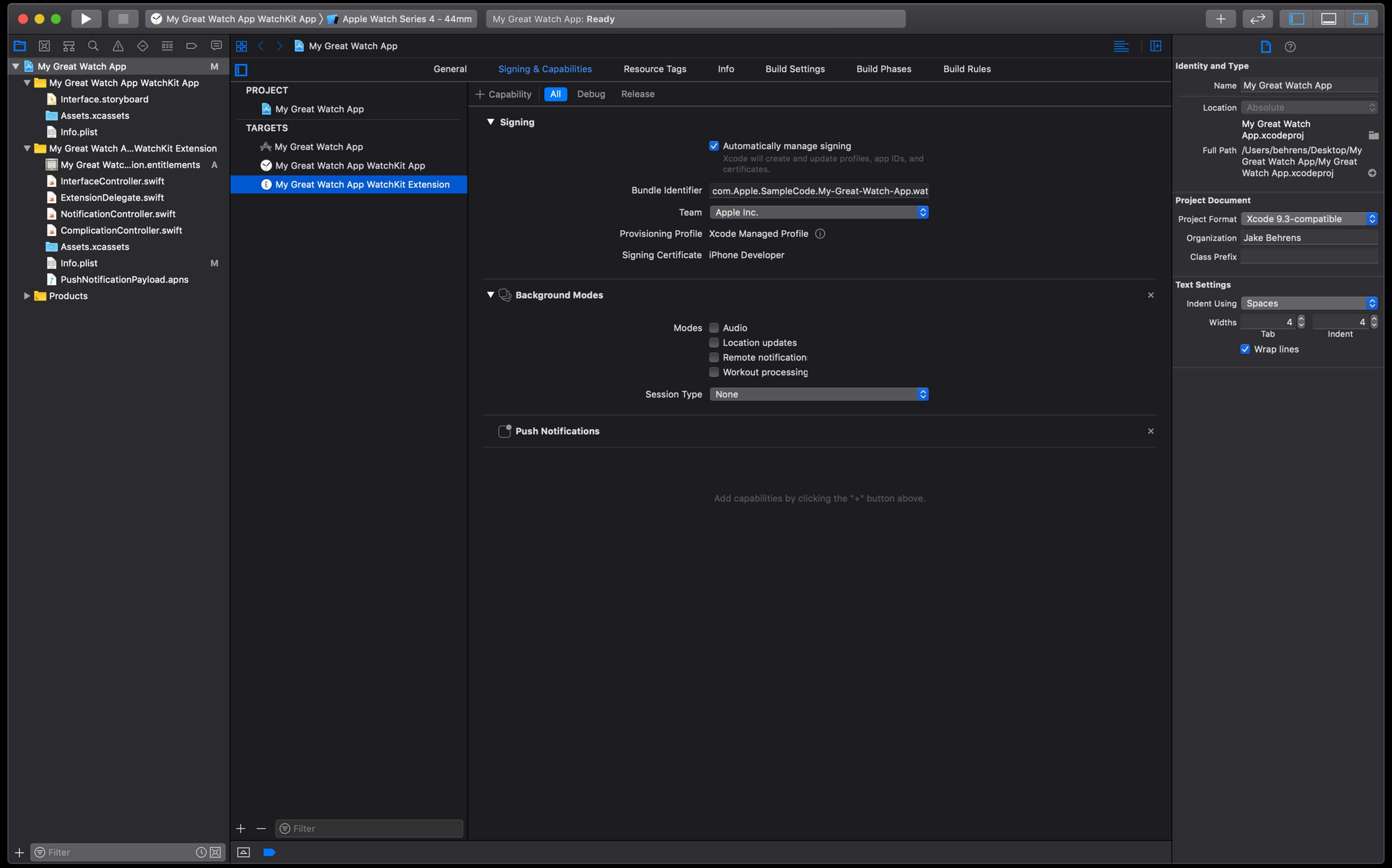
Notification service extension support

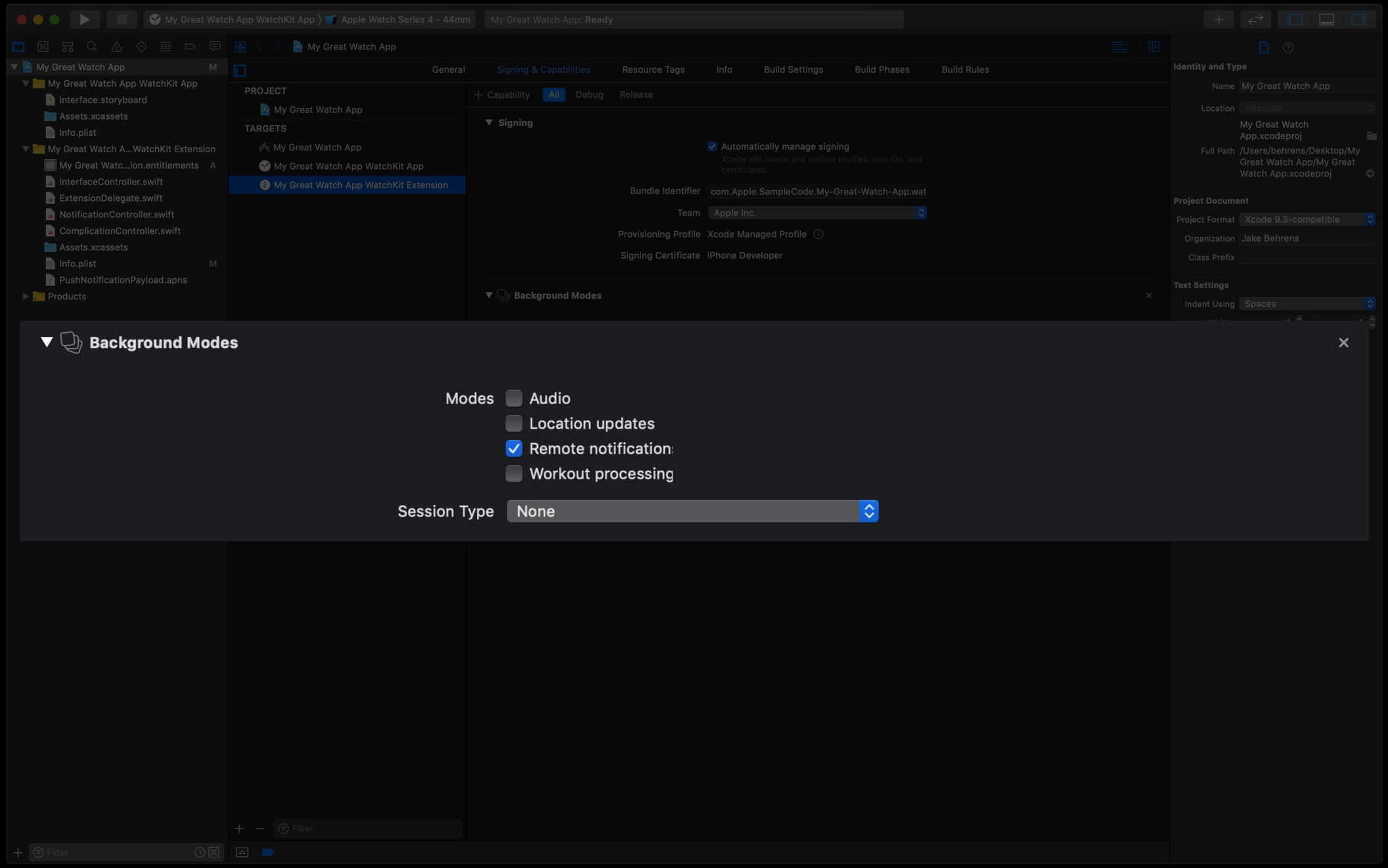

My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
Filter
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Filter
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Signing Certificate
iPhone Developer
Add capabilities by clicking the "+" button above.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
4
Tab
Indent
Wrap lines


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Push Notifications
Capabilities
Media
Push Notifications
Push Notifications
Apple Push Notification service is a robust and highly efficient service for propagating information to devices.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
4
Tab
Indent
Wrap lines
Filter
Filter


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
My Great Watc...ion.entitlements
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
Filter
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Signing Certificate
iPhone Developer
Push Notifications
Add capabilities by clicking the "+" button above.
Filter
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
Tab
4
Indent
Wrap lines


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Background Modes
Capabilities
Media
Background Modes
Background Modes
Background Modes specifies that the app provides specific background services and must be allowed to continue running while in the background. These keys should be used sparingly and only by apps providing the indicated services. Where alternatives for running in the background exist, those alternatives should be used instead.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
Tab
Indent
Wrap lines


My Great Watch App WatchKit App 〉 Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
My Great Watch App
My Great Watch App WatchKit App
Interface.storyboard
Assets.xcassets
Info.plist
My Great Watch A...WatchKit Extension
My Great Watc...ion.entitlements
InterfaceController.swift
ExtensionDelegate.swift
NotificationController.swift
ComplicationController.swift
Assets.xcassets
Info.plist
PushNotificationPayload.apns
Products
Filter
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Filter
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Signing Certificate
iPhone Developer
Background Modes
Modes
Audio
Location updates
Remote notification
Workout processing
Session Type
None
Push Notifications
Add capabilities by clicking the "+" button above.
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Widths
4
Tab
4
Indent
Wrap lines


My Great Watch App WatchKit App
Apple Watch Series 4 - 44mm
My Great Watch App: Ready
My Great Watch App
General
Signing & Capabilities
Resource Tags
Info
Build Settings
Build Phases
Build Rules
PROJECT
My Great Watch App
TARGETS
My Great Watch App
My Great Watch App WatchKit App
My Great Watch App WatchKit Extension
Capability
All
Debug
Release
Signing
Automatically manage signing
Xcode will create and update profiles, app IDs, and certificates.
Bundle Identifier
com.Apple.SampleCode.My-Great-Watch-App.wat
Team
Apple Inc.
Provisioning Profile
Xcode Managed Profile
Signing Certificate
iPhone Developer
Background Modes
Identity and Type
Name
My Great Watch App
Location
Absolute
My Great Watch App.xcodeproj
Full Path
/Users/behrens/Desktop/My Great Watch App/My Great Watch App.xcodeproj
Project Document
Project Format
Xcode 9.3-compatible
Organization
Jake Behrens
Class Prefix
Text Settings
Indent Using
Spaces
Modes
Audio
Location updates
Remote notification
Workout processing
Session Type
None

```swift
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func applicationDidFinishLaunching() {
        let center = UNUserNotificationCenter.current()
        center.requestAuthorization(options: [.alert, .sound]) { (granted, error) in
            // Enable or disable features based on authorization.
            if (granted) {
                WKExtension.shared().registerForRemoteNotifications()
            } else { /* Handle no access */ }
        }
    }
```

```
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func applicationDidFinishLaunching() {
        let center = UNUserNotificationCenter.current()
        center.requestAuthorization(options: [.alert, .sound]) { (granted, error) in
            // Enable or disable features based on authorization.
            if (granted) {
                WKExtension.shared().registerForRemoteNotifications()
            } else { /* Handle no access */ }
        }
    }
```

```swift
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func applicationDidFinishLaunching() {
        let center = UNUserNotificationCenter.current()
        center.requestAuthorization(options: [.alert, .sound]) { (granted, error) in
            // Enable or disable features based on authorization.
            if (granted) {
                WKExtension.shared().registerForRemoteNotifications()
            } else { /* Handle no access */ }
        }
    }
```

```swift
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didRegisterForRemoteNotifications(withDeviceToken deviceToken: Data) {
        /* Forward the token to your provider, using a custom method. */
    }

    func didFailToRegisterForRemoteNotificationsWithError(_ error: Error) {
        /* Disable remote notification features */
    }
```

```
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didRegisterForRemoteNotifications(withDeviceToken deviceToken: Data) {
        /* Forward the token to your provider, using a custom method. */
    }

    func didFailToRegisterForRemoteNotificationsWithError(_ error: Error) {
        /* Disable remote notification features */
    }
```

```
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didRegisterForRemoteNotifications(withDeviceToken deviceToken: Data) {
        /* Forward the token to your provider, using a custom method. */
    }

    func didFailToRegisterForRemoteNotificationsWithError(_ error: Error) {
        /* Disable remote notification features */
    }
```

```
// WatchKit code for registration of notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didRegisterForRemoteNotifications(withDeviceToken deviceToken: Data) {
        /* Forward the token to your provider, using a custom method. */
    }

    func didFailToRegisterForRemoteNotificationsWithError(_ error: Error) {
        /* Disable remote notification features */
    }
```

```
// WatchKit code for handling background notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
                                      fetchCompletionHandler: @escaping
                                      (WKBackgroundFetchResult) -> Void) {
        /* Handle background notification */
        fetchCompletionHandler(.newData)
    }
```

```
// WatchKit code for handling background notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
                                      fetchCompletionHandler: @escaping
                                      (WKBackgroundFetchResult) -> Void) {
        /* Handle background notification */
        fetchCompletionHandler(.newData)
    }
```

```
// WatchKit code for handling background notifications

import WatchKit
import UserNotifications

class ExtensionDelegate: NSObject, WKExtensionDelegate {

    func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
                                      fetchCompletionHandler: @escaping
                                      (WKBackgroundFetchResult) -> Void) {
        /* Handle background notification */
        fetchCompletionHandler(.newData)
    }
```

Details

Details

`apns-push-type` is required

Details

`apns-push-type` is required

`apns-topic` is your WatchKit app bundle identifier

Details

`apns-push-type` is required

`apns-topic` is your WatchKit app bundle identifier

Alert Coordination for duplicate notifications when sent simultaneously

Complication Pushes on watchOS

Complication Pushes on watchOS

Update your app if complication is enabled on the active watch face

Complication Pushes on watchOS

NEW

Update your app if complication is enabled on the active watch face

PushKit now available on watchOS

Complication Pushes on watchOS

NEW

Update your app if complication is enabled on the active watch face

PushKit now available on watchOS

Registration and delivery

```swift
import PushKit

func registerForComplicationPushes() {
    let pushRegistry = PKPushRegistry(queue: .main)
    pushRegistry.delegate = self
    pushRegistry.desiredPushTypes = [.complication]
}

func pushRegistry(_ registry: PKPushRegistry,
                  didUpdate pushCredentials: PKPushCredentials,
                  for type: PKPushType) {
    /* Forward complication token to server */
}

func pushRegistry(_ registry: PKPushRegistry, didInvalidatePushTokenFor type: PKPushType) {
    /* Handle invalidated token */
}
```

```swift
import PushKit

func registerForComplicationPushes() {
    let pushRegistry = PKPushRegistry(queue: .main)
    pushRegistry.delegate = self
    pushRegistry.desiredPushTypes = [.complication]
}

func pushRegistry(_ registry: PKPushRegistry,
                  didUpdate pushCredentials: PKPushCredentials,
                  for type: PKPushType) {
    /* Forward complication token to server */
}

func pushRegistry(_ registry: PKPushRegistry, didInvalidatePushTokenFor type: PKPushType) {
    /* Handle invalidated token */
}
```

```swift
import PushKit

func registerForComplicationPushes() {
    let pushRegistry = PKPushRegistry(queue: .main)
    pushRegistry.delegate = self
    pushRegistry.desiredPushTypes = [.complication]
}

func pushRegistry(_ registry: PKPushRegistry,
                  didUpdate pushCredentials: PKPushCredentials,
                  for type: PKPushType) {
    /* Forward complication token to server */
}

func pushRegistry(_ registry: PKPushRegistry, didInvalidatePushTokenFor type: PKPushType) {
    /* Handle invalidated token */
}
```

```swift
import PushKit

func registerForComplicationPushes() {
    let pushRegistry = PKPushRegistry(queue: .main)
    pushRegistry.delegate = self
    pushRegistry.desiredPushTypes = [.complication]
}

func pushRegistry(_ registry: PKPushRegistry,
                  didUpdate pushCredentials: PKPushCredentials,
                  for type: PKPushType) {
    /* Forward complication token to server */
}

func pushRegistry(_ registry: PKPushRegistry, didInvalidatePushTokenFor type: PKPushType) {
    /* Handle invalidated token */
}
```

```
// Handle incoming complication push

import PushKit

func pushRegistry(_ registry: PKPushRegistry,
                  didReceiveIncomingPushWith payload: PKPushPayload,
                  for type: PKPushType,
                  completion: @escaping () -> Void) {
    /* Handle receiving complication push
       Reload complication timeline */
}
```

```swift
// Handle incoming complication push

import PushKit

func pushRegistry(_ registry: PKPushRegistry,
                  didReceiveIncomingPushWith payload: PKPushPayload,
                  for type: PKPushType,
                  completion: @escaping () -> Void) {
    /* Handle receiving complication push
       Reload complication timeline */
}
```





Networking

Networking

URLSession

Networking

URLSession

CloudKit

URLSession

URLSession

Migrate all WatchConnectivity usage to URLSession

URLSession

Migrate all WatchConnectivity usage to URLSession

Background sessions

WatchConnectivity

WatchConnectivity

Use WatchConnectivity for companion app specific interactions

WatchConnectivity

Use WatchConnectivity for companion app specific interactions

```
open var isCompanionAppInstalled: Bool { get }
```

CloudKit

CloudKit

NEW

CKSubscription

CloudKit

NEW

CKSubscription

CloudKit notifications

CloudKit

NEW

CKSubscription

CloudKit notifications

CKSubscriptions

CKSubscriptions

Subscribe to changes

CKSubscriptions

Subscribe to changes

Pushes tell you when to update

CKSubscriptions

Subscribe to changes

Pushes tell you when to update

Retrieve only what has changed


9:41
FaceTime
Monday
3
Calendar
Photos
Camera
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Health
Home
Wallet
Settings

Data


Mon
3

Data


9:41
FaceTime
Monday
3
Calendar
Photos
Camera
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Health
Home
Wallet
Settings

Data

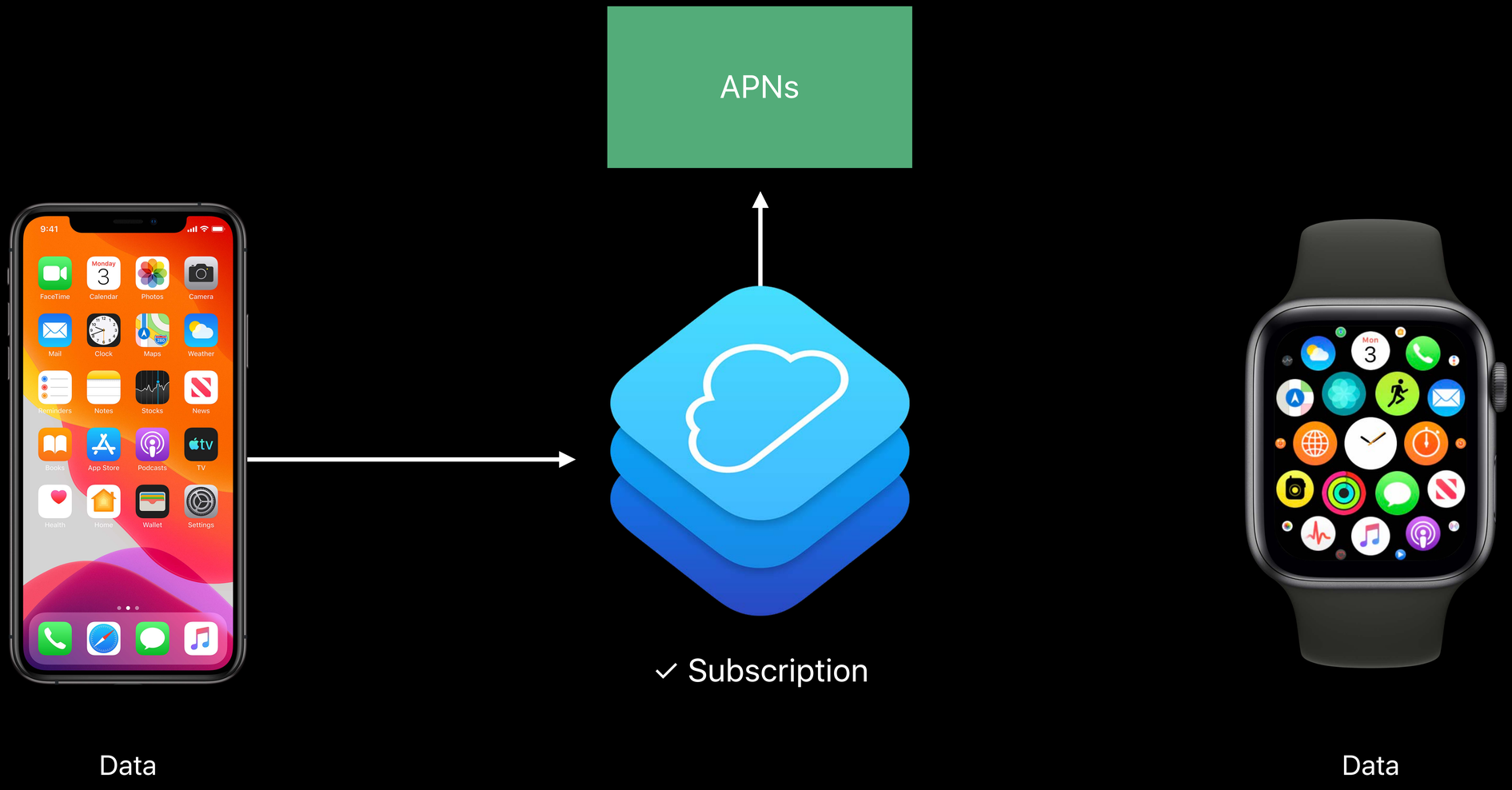

Mon
3

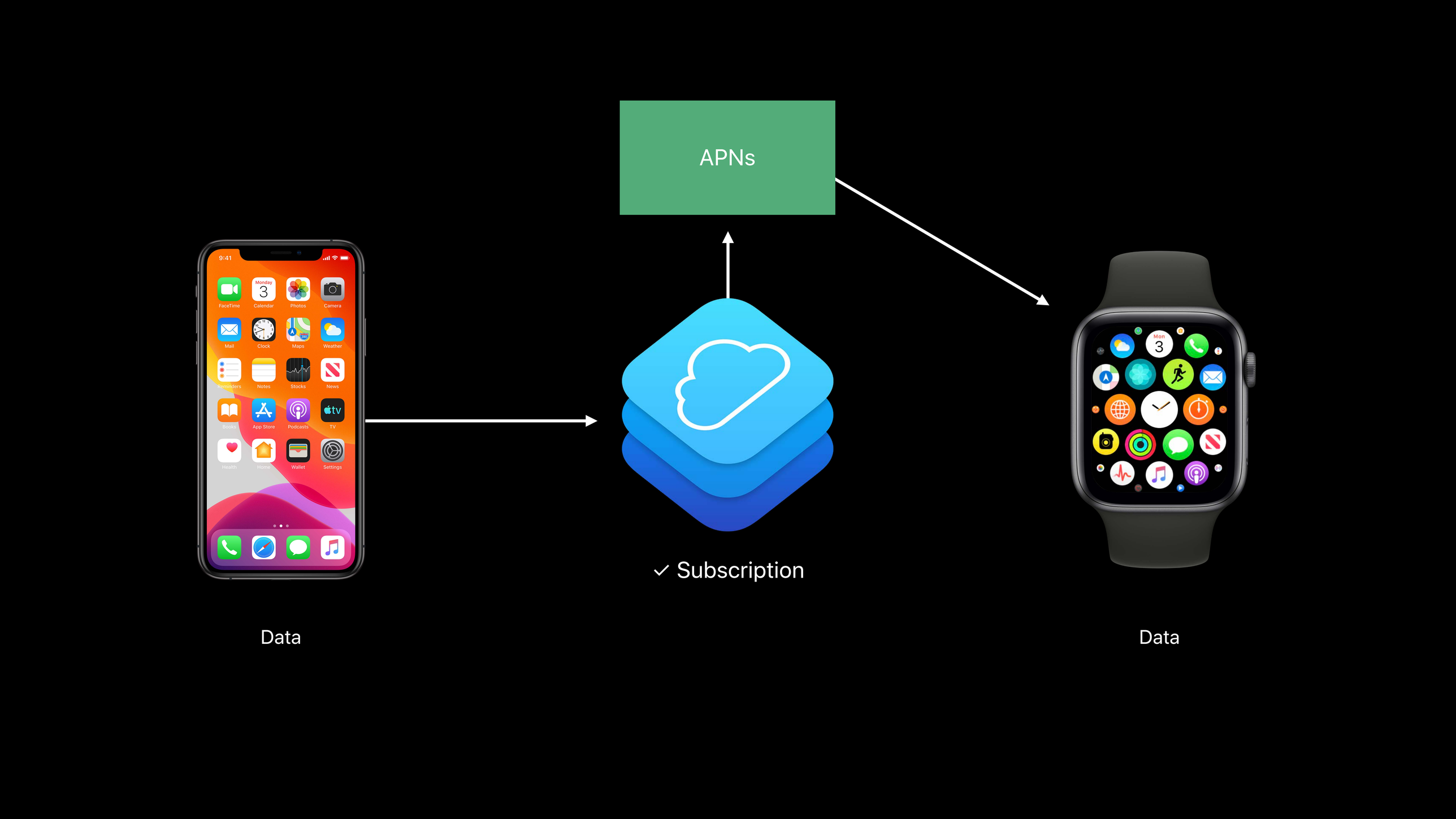
Data

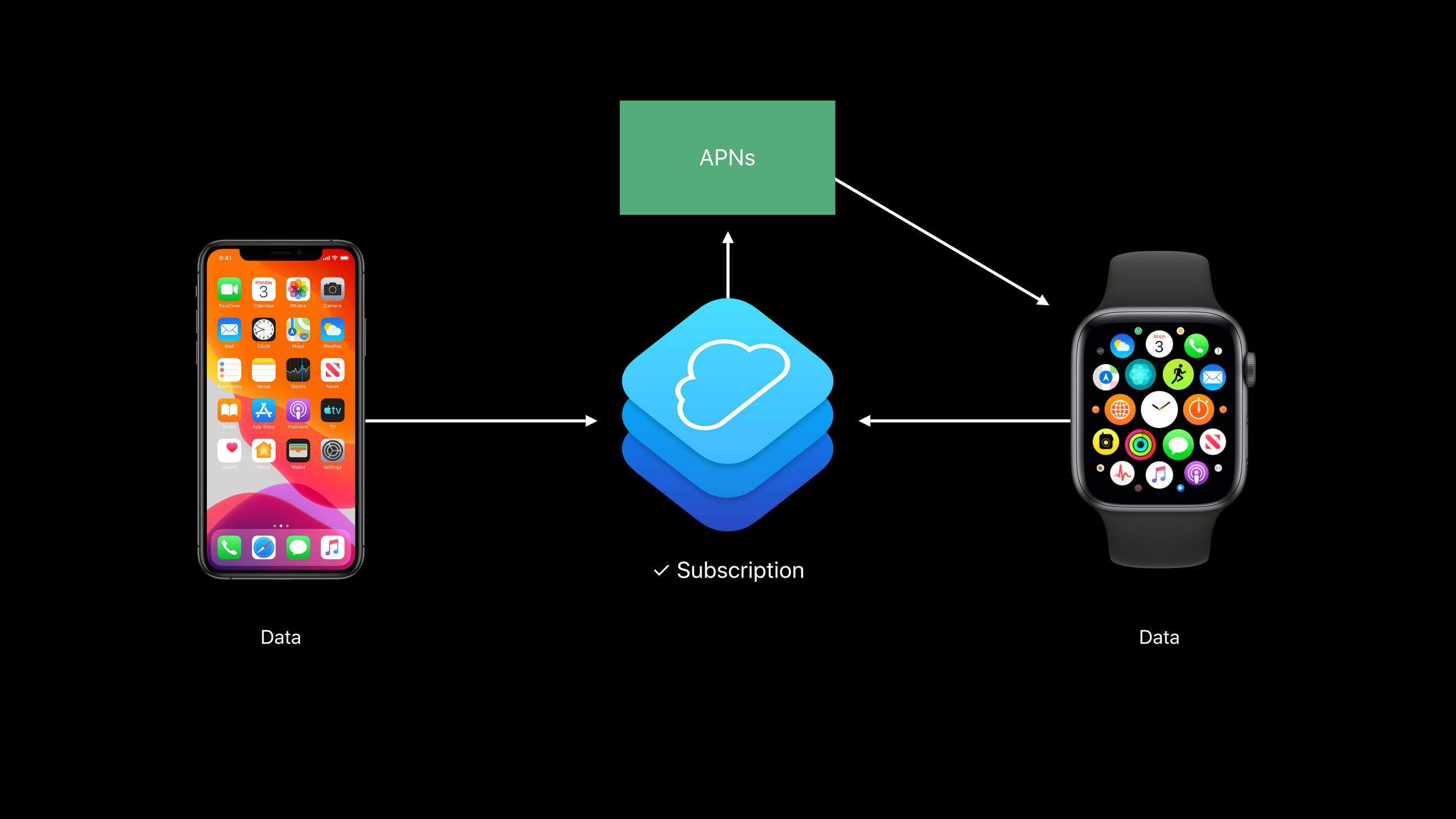
Data

✓ Subscription

Data


APNs
9:41
FaceTime
Calendar
Photos
Camera
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Health
Home
Wallet
Settings
✓ Subscription
Data
Data


APNs
9:41
FaceTime
Calendar
Photos
Camera
Mail
Clock
Maps
Weather
Reminders
Notes
Stocks
News
Books
App Store
Podcasts
TV
Health
Home
Wallet
Settings
✓ Subscription
Data
Data


APNs
✓ Subscription
Data
Data

```
// Background push

let notificationInfo = CKSubscription.NotificationInfo()

// Set only this property

notificationInfo.shouldSendContentAvailable = true

// CloudKit will deliver a push, listen for pushes via:

func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
   fetchCompletionHandler: @escaping (WKBackgroundFetchResult) -> Void) {…}
```

```
// Background push

let notificationInfo = CKSubscription.NotificationInfo()

// Set only this property

notificationInfo.shouldSendContentAvailable = true

// CloudKit will deliver a push, listen for pushes via:

func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
   fetchCompletionHandler: @escaping (WKBackgroundFetchResult) -> Void) {…}
```

```
// Background push

let notificationInfo = CKSubscription.NotificationInfo()

// Set only this property

notificationInfo.shouldSendContentAvailable = true

// CloudKit will deliver a push, listen for pushes via:

func didReceiveRemoteNotification(_ userInfo: [AnyHashable : Any],
   fetchCompletionHandler: @escaping (WKBackgroundFetchResult) -> Void) {…}
```



Summary

Summary

Freedom and independence

Summary

Freedom and independence

System and developer capabilities

Summary

Freedom and independence

System and developer capabilities

Make your apps independent

More Information

developer.apple.com/wwdc19/208

SwiftUI on watchOS	Wednesday, 2:00
watchOS Independence Lab	Wednesday, 9:00


WWDC19