App Frameworks #WWDC16

Increase Usage of Your App with Proactive Suggestions

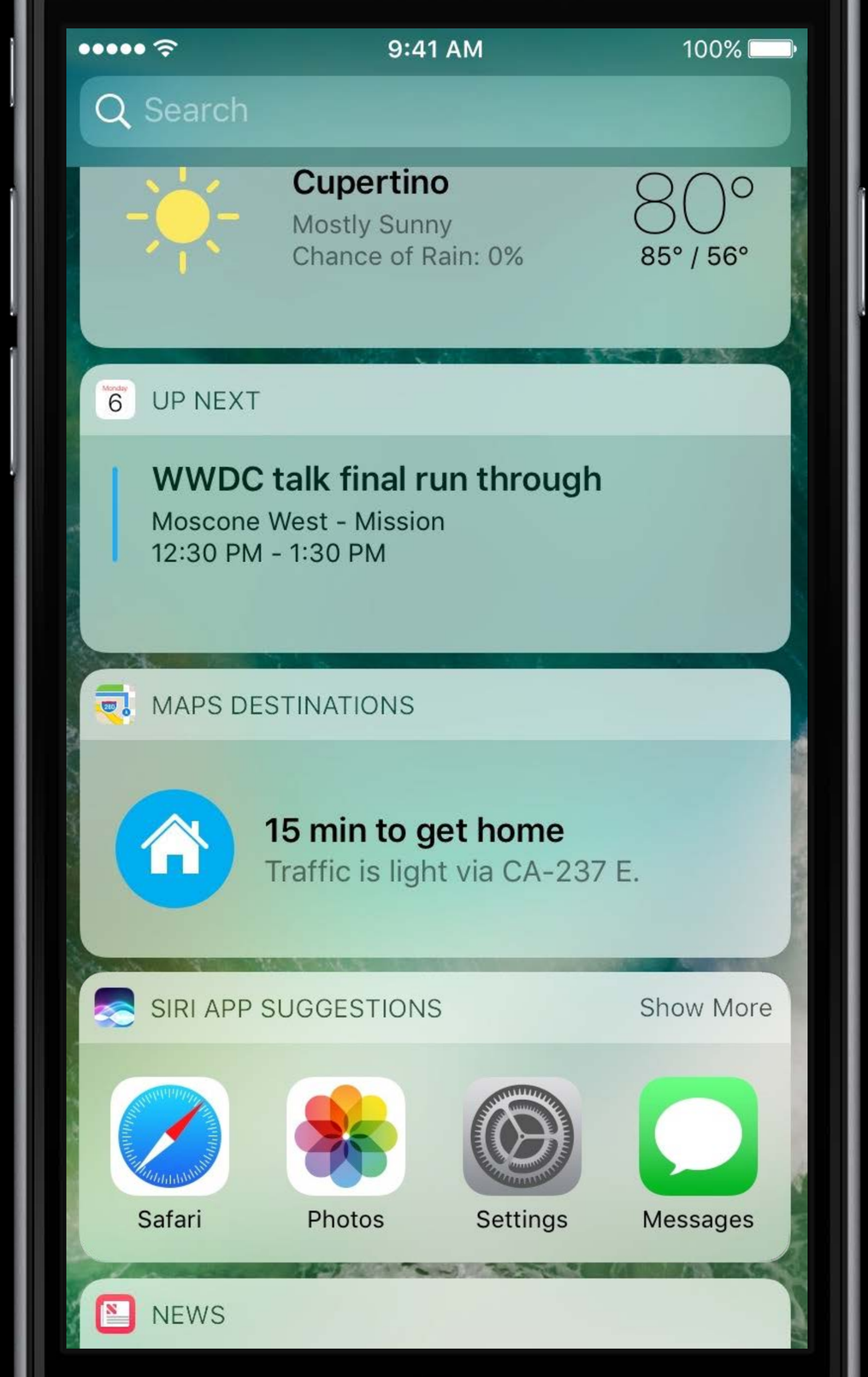
Session 240

Daniel Gross Siri
Sofiane Toudji Software Engineering

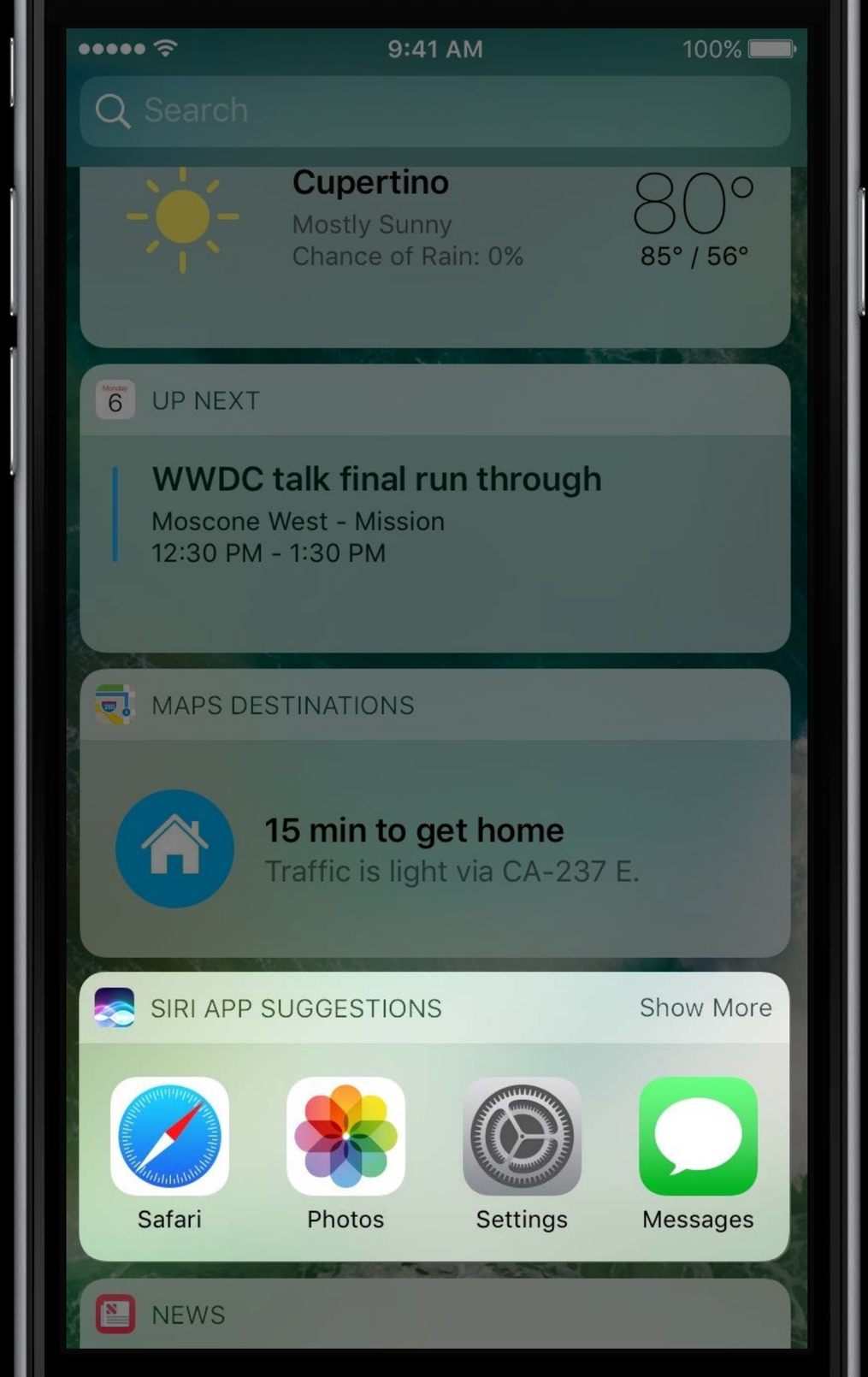
You want more users.

You want more users.

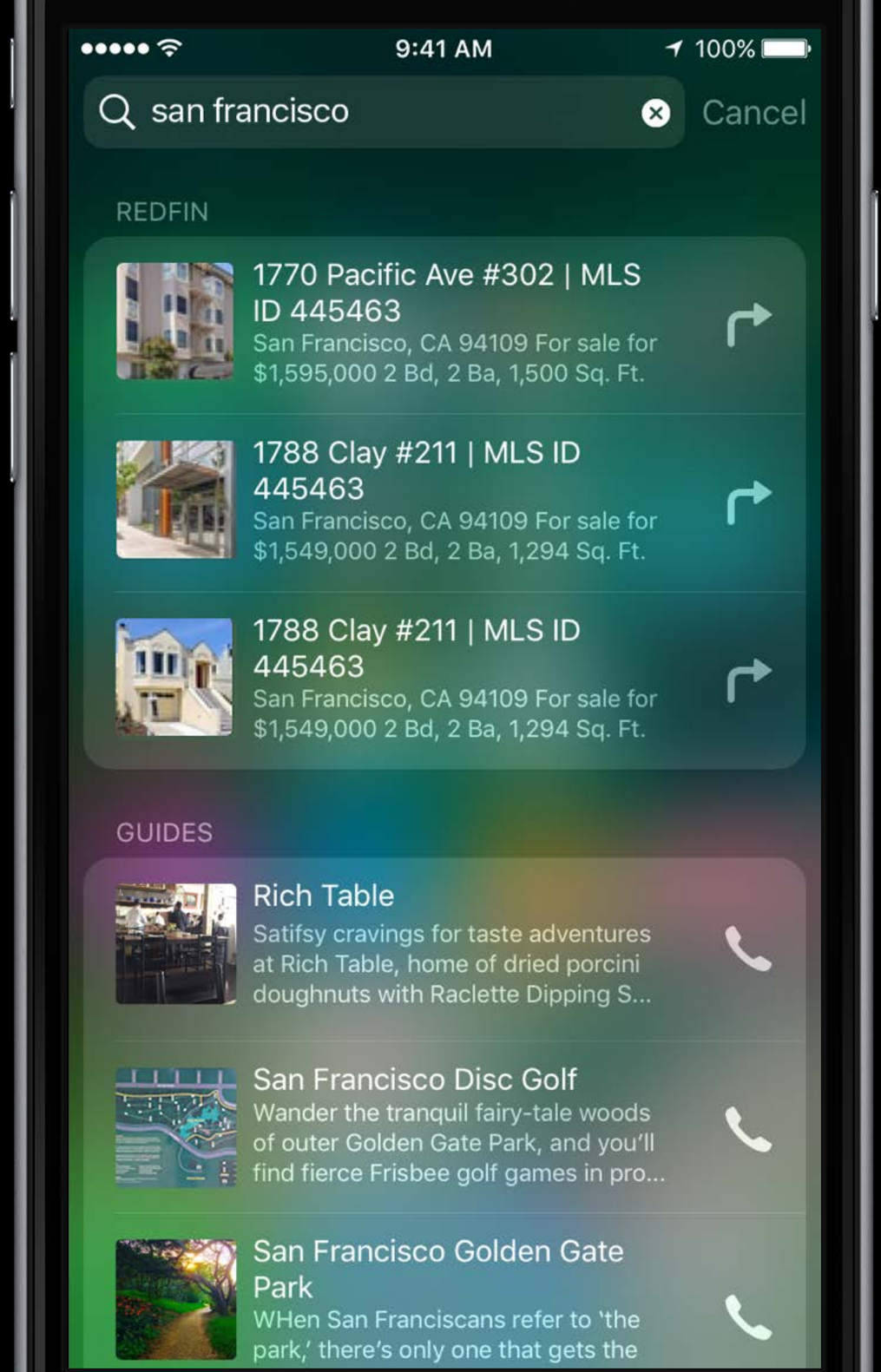
Apple wants to offer your app at the right moment.



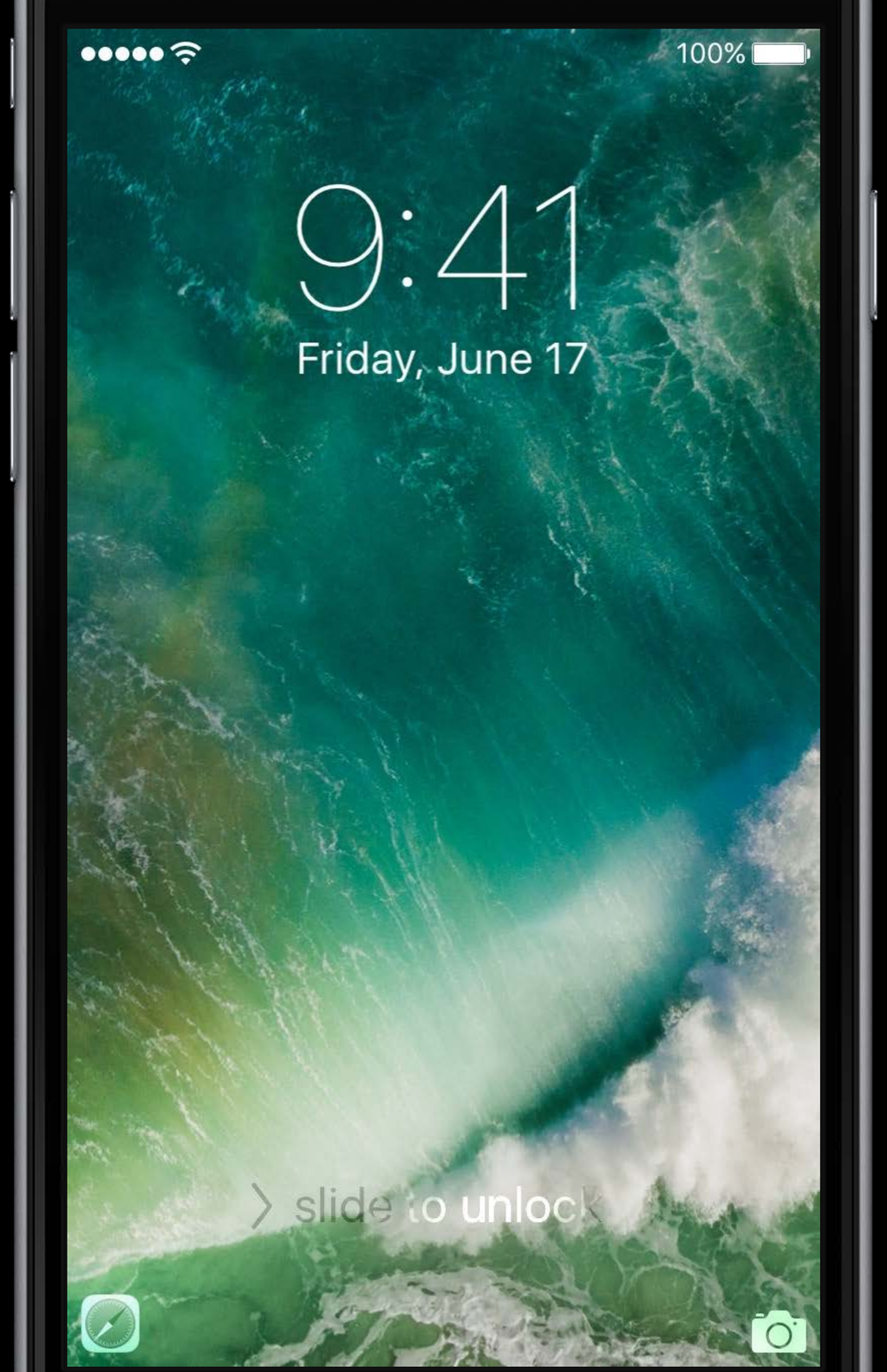
Siri App Suggestions



Siri App Suggestions



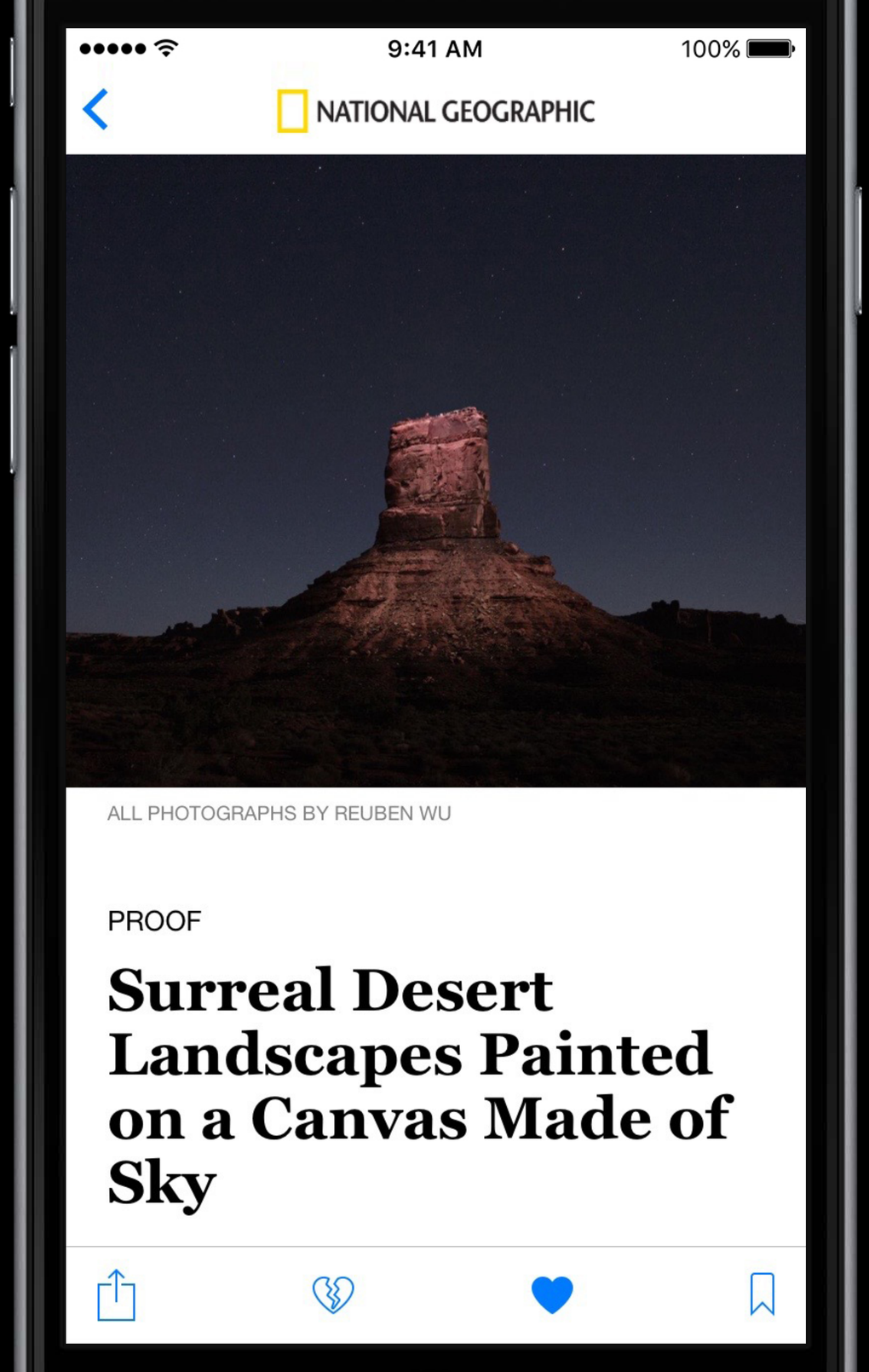
Spotlight Search



Handoff

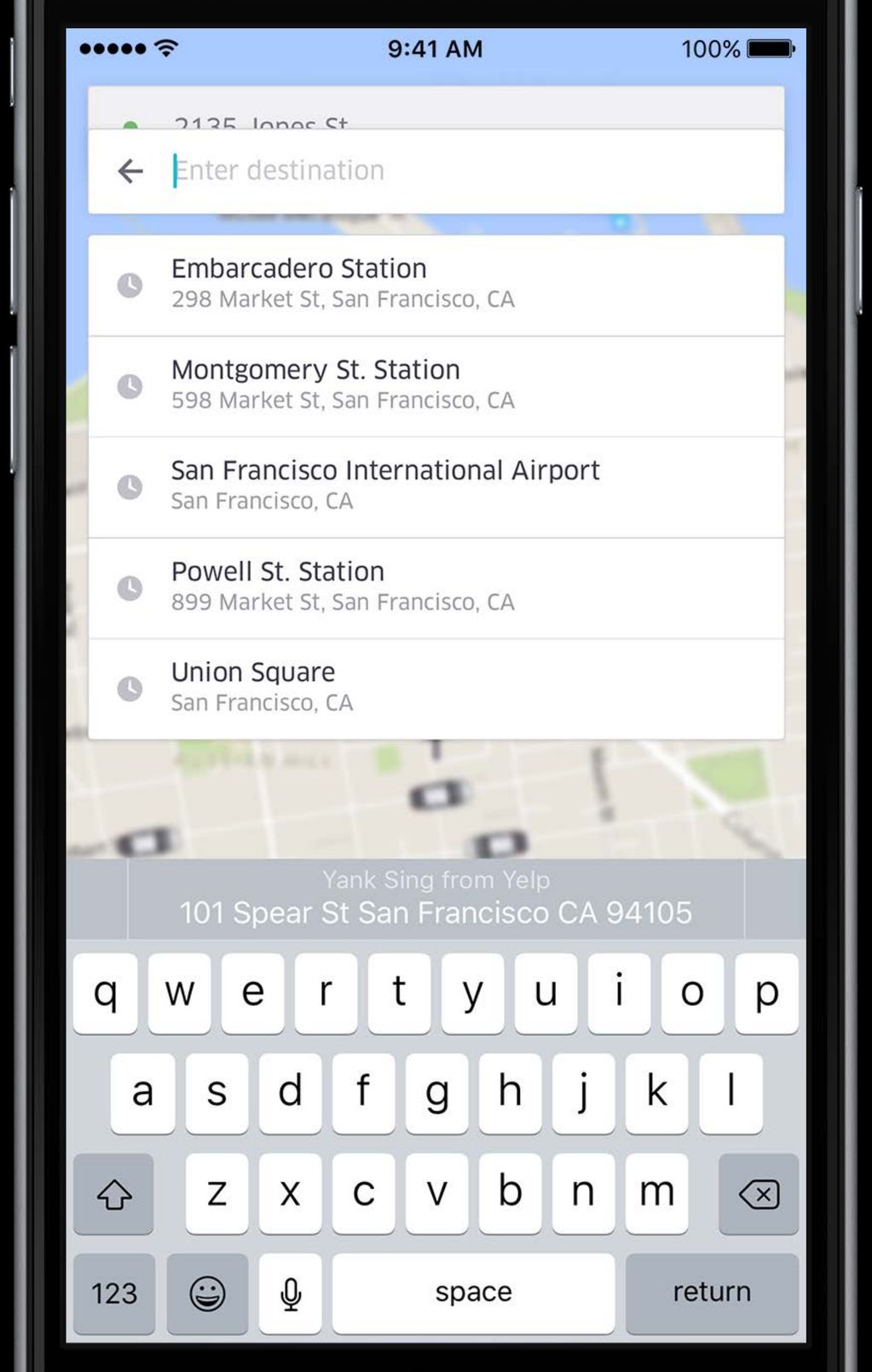

9:41 AM
100%
3:29
-56:31
The Power Of Design
TED Radio Hour — May 19, 2016 at
15
15
npr
TED
RADIO
HOUR
Press home to open

Now Playing



Contextual Siri Reminders

Contextual Siri Reminders


9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA
Yank Sing from Yelp
101 Spear St San Francisco CA 94105
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

QuickType

Multitasking

Contact Interactions

Proactive Suggestions

Overview

Proactive Suggestions

Overview

To learn when to promote you, the OS needs to know more about your data or app

Proactive Suggestions

Overview

To learn when to promote you, the OS needs to know more about your data or app

A few simple APIs deeply integrate your app into iOS and macOS

Proactive Suggestions

Overview

To learn when to promote you, the OS needs to know more about your data or app

A few simple APIs deeply integrate your app into iOS and macOS

Deeper integration with Siri “for free”

Proactive Suggestions

Agenda

Proactive Suggestions

Agenda

NSUserActivity and schema.org

Proactive Suggestions

Agenda

NSUserActivity and schema.org

Location Suggestions

Proactive Suggestions

Agenda

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Proactive Suggestions

Agenda

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Summary

Proactive Suggestions

Agenda

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Summary

1. Handoff
2. Spotlight Search
3. Contextual Siri Reminders
4. Location Suggestions
5. Contextual Siri Requests
6. Contact Interactions

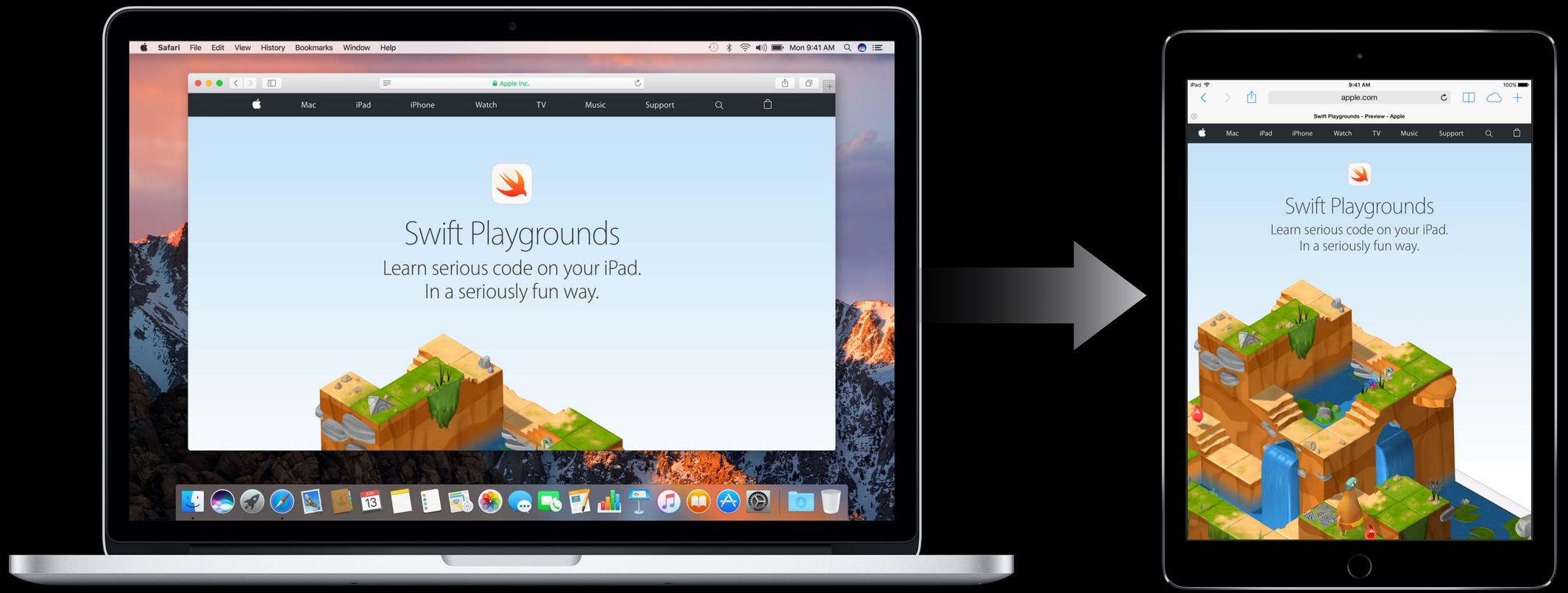
NSUserActivity

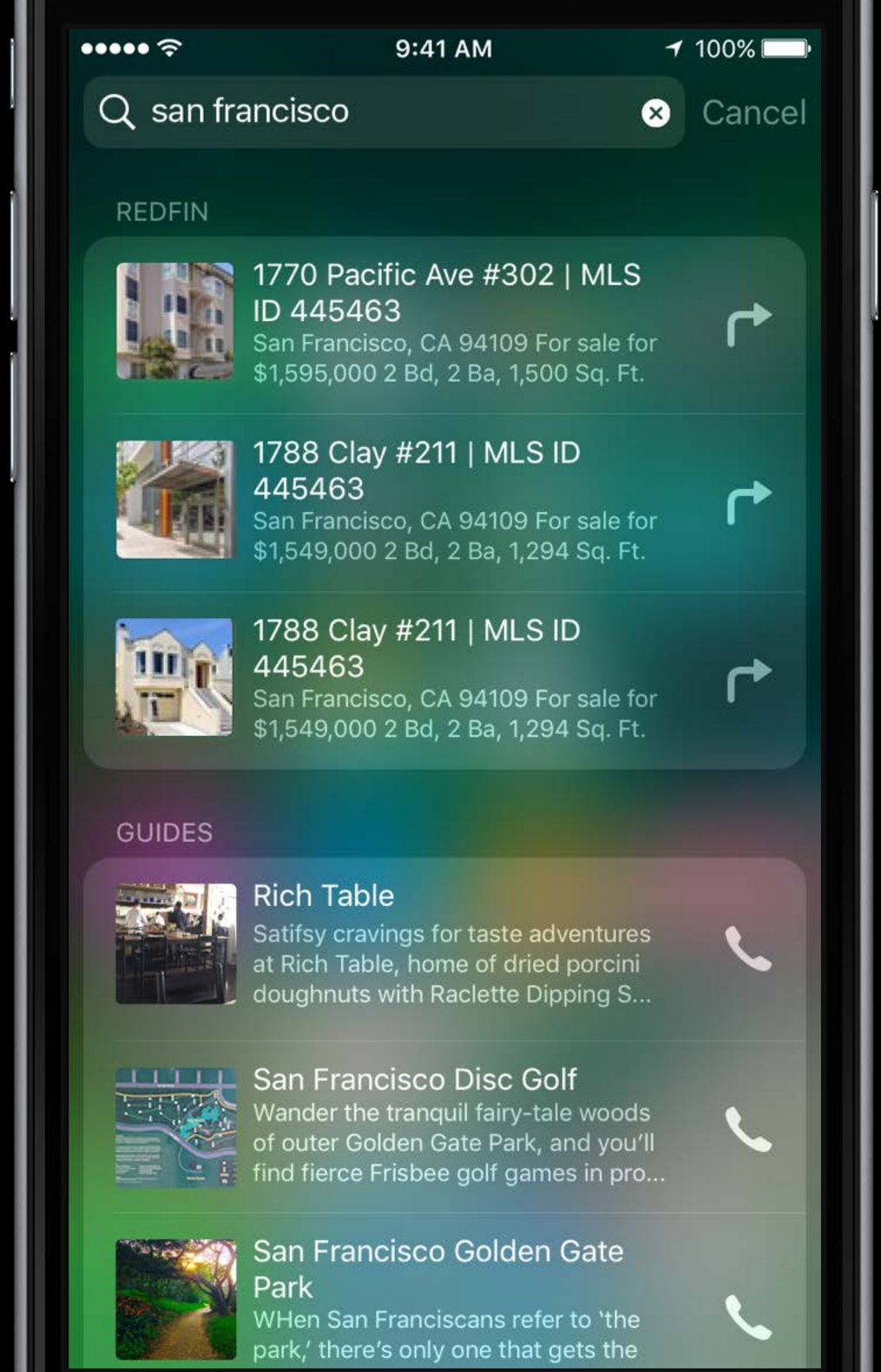
NSUserActivity

Handoff

NSUserActivity

Handoff

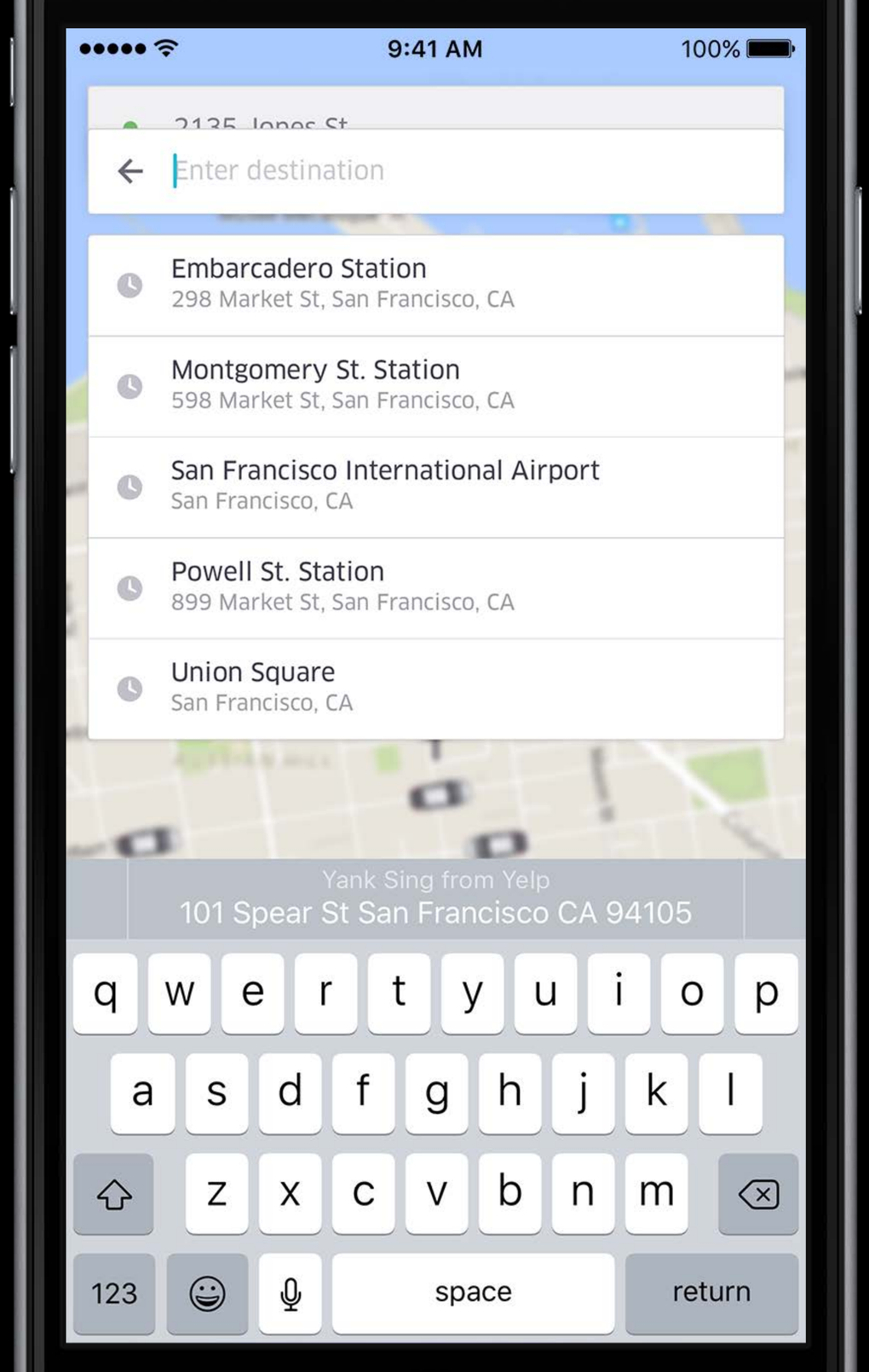


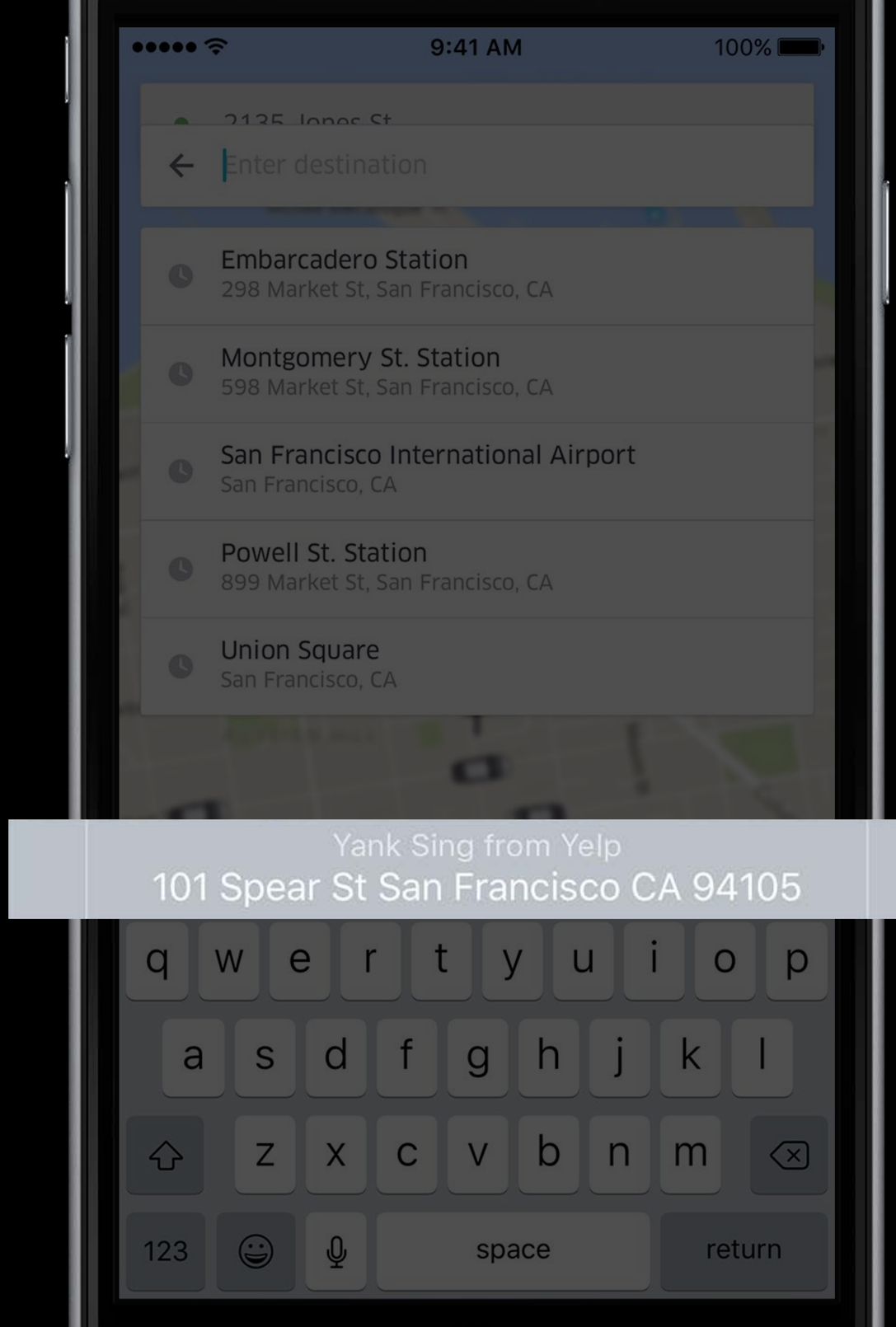


Spotlight Search

NEW

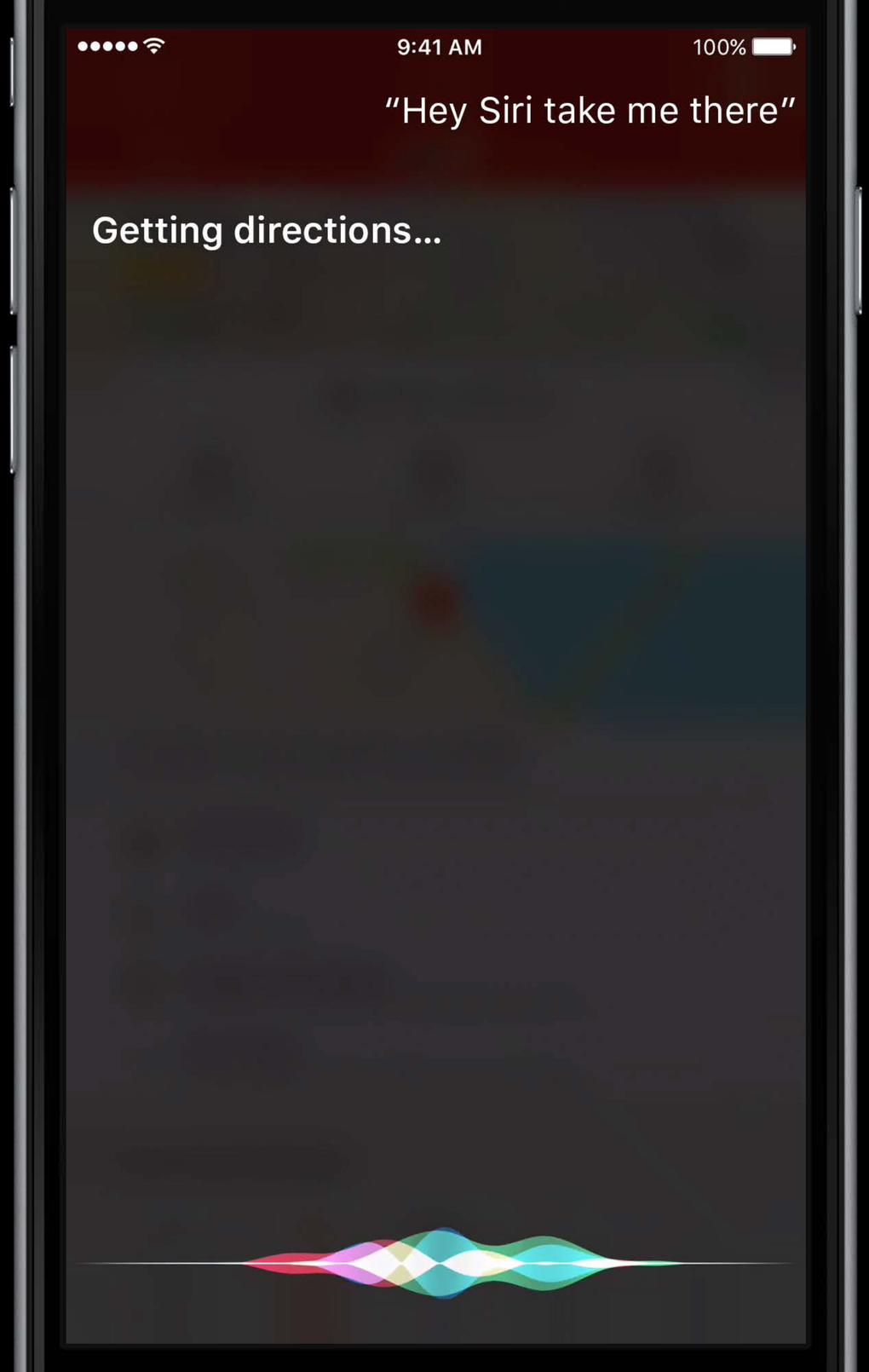
Location Suggestions





NEW

Location Suggestions



NEW

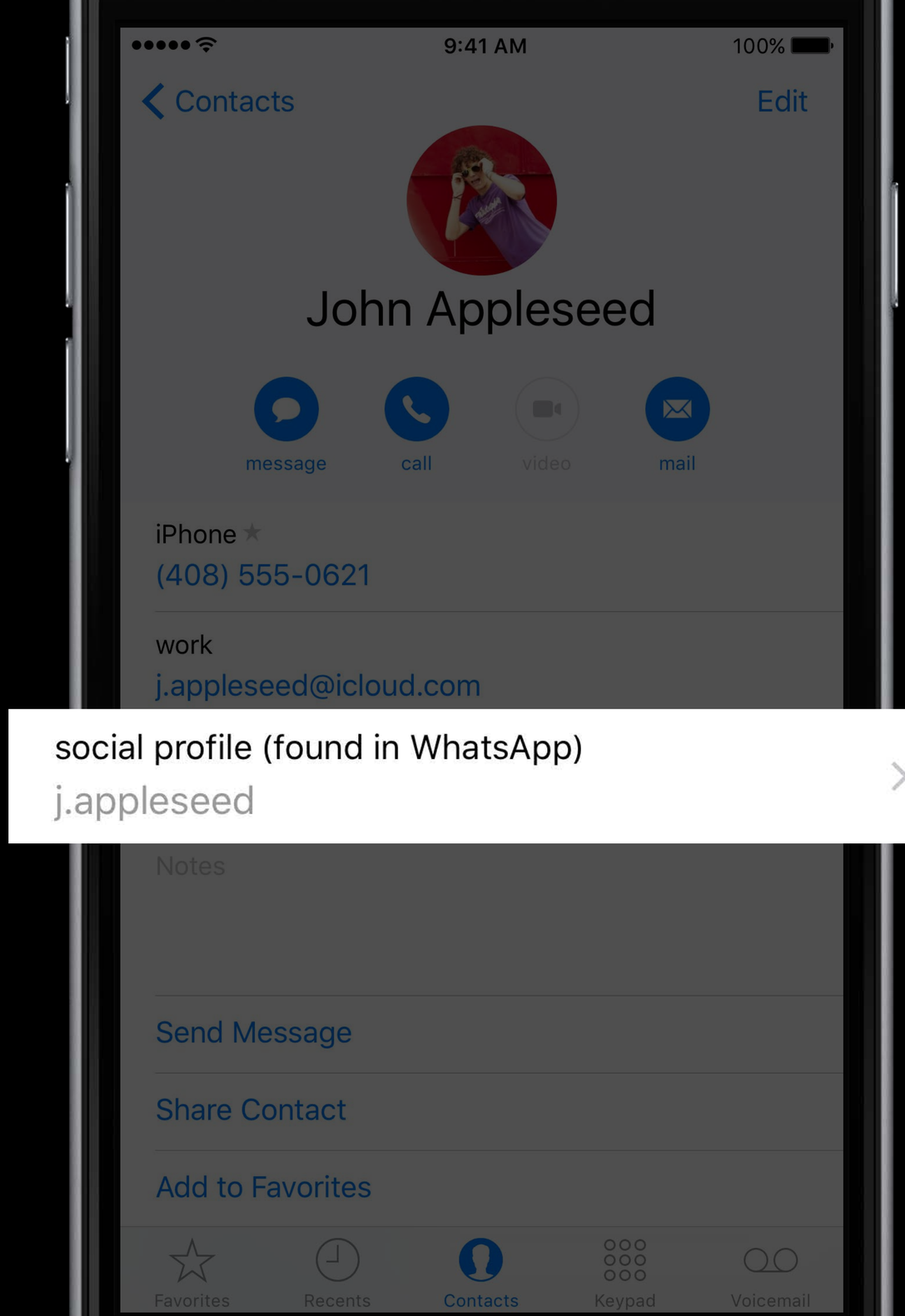
Contextual Siri Requests

NEW

Contact Interactions

NEW

Contact Interactions



How does it work?

NSUserActivity

Native apps

schema.org

Web apps

NSUserActivity

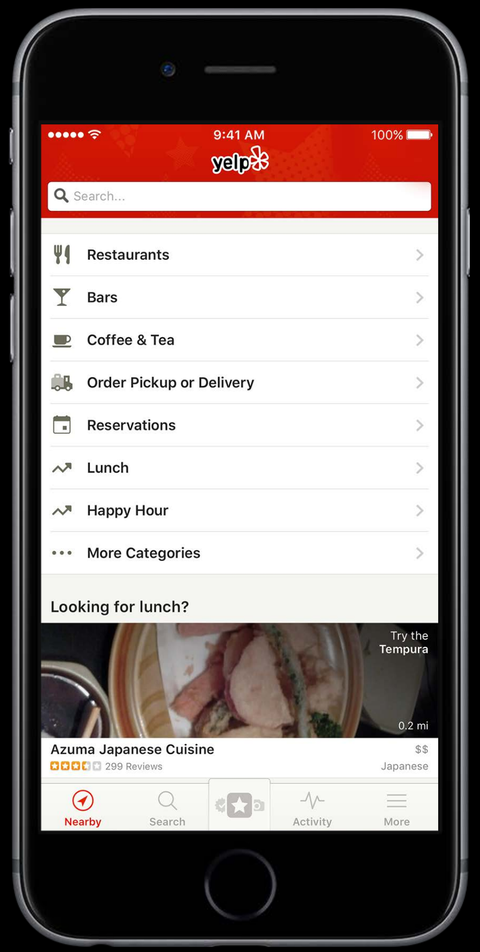
Native apps

schema.org

Web apps

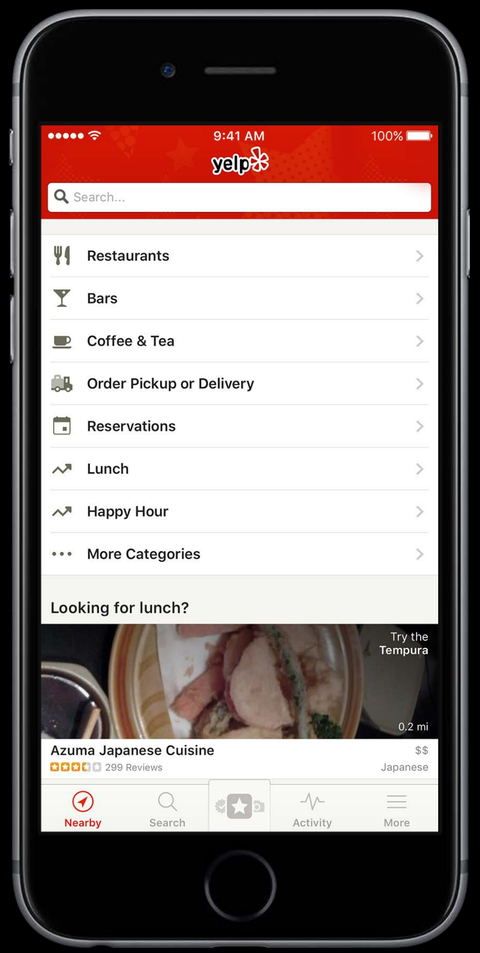
NSUserActivity

Capturing app state



NSUserActivity

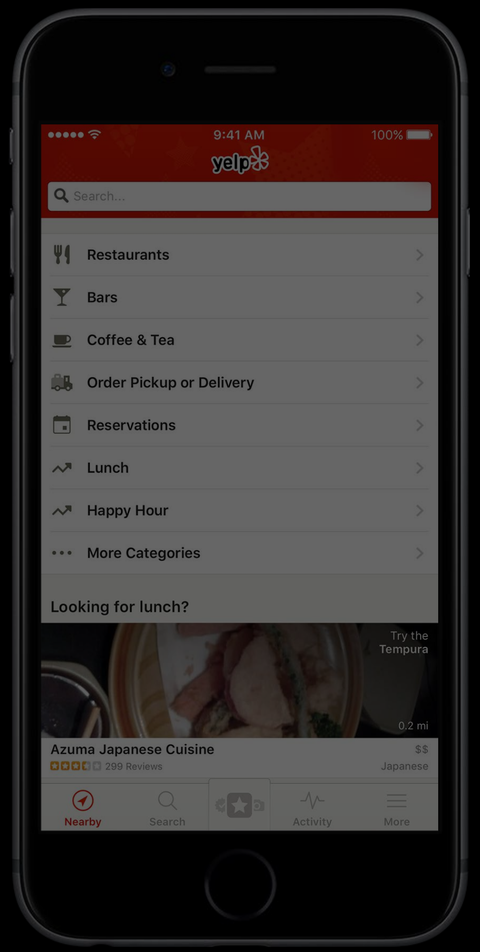
Capturing app state

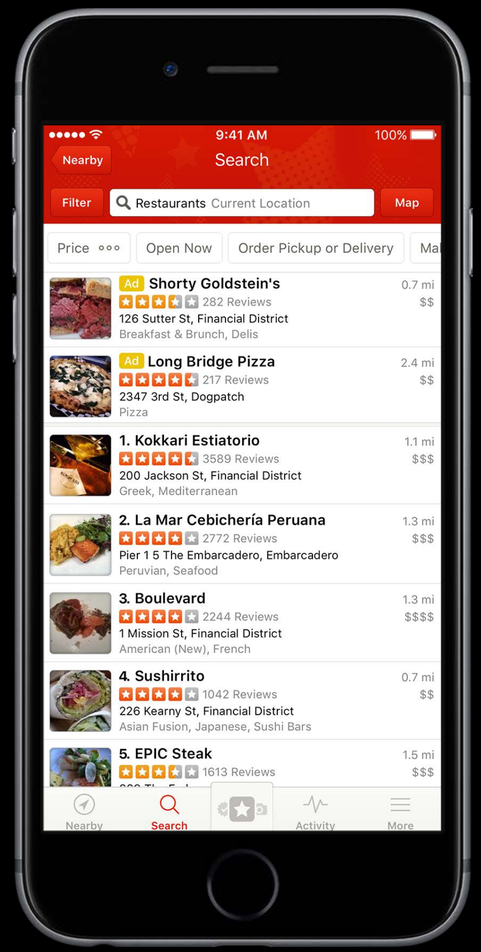


NSUserActivity

NSUserActivity

Capturing app state

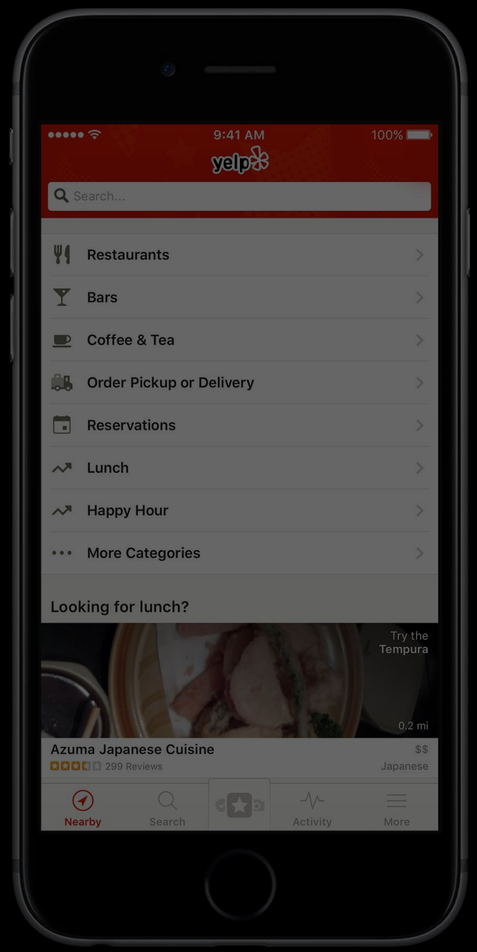




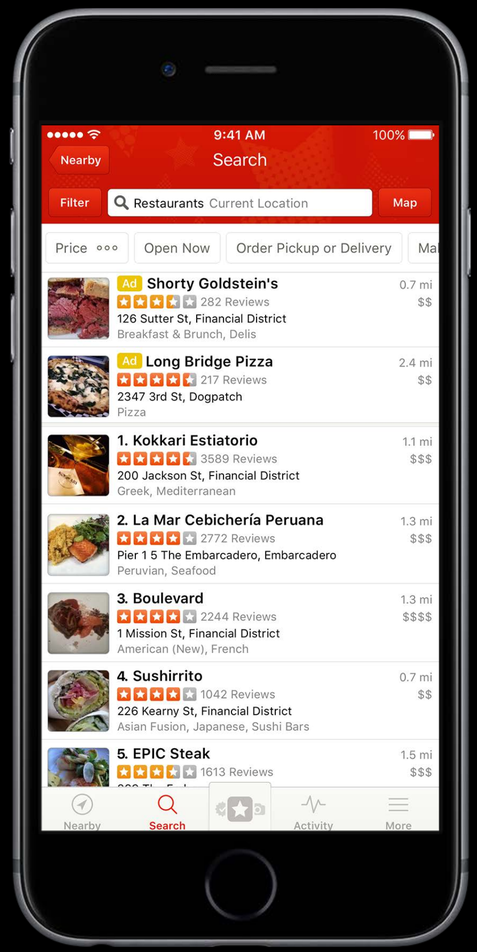
NSUserActivity

NSUserActivity

Capturing app state



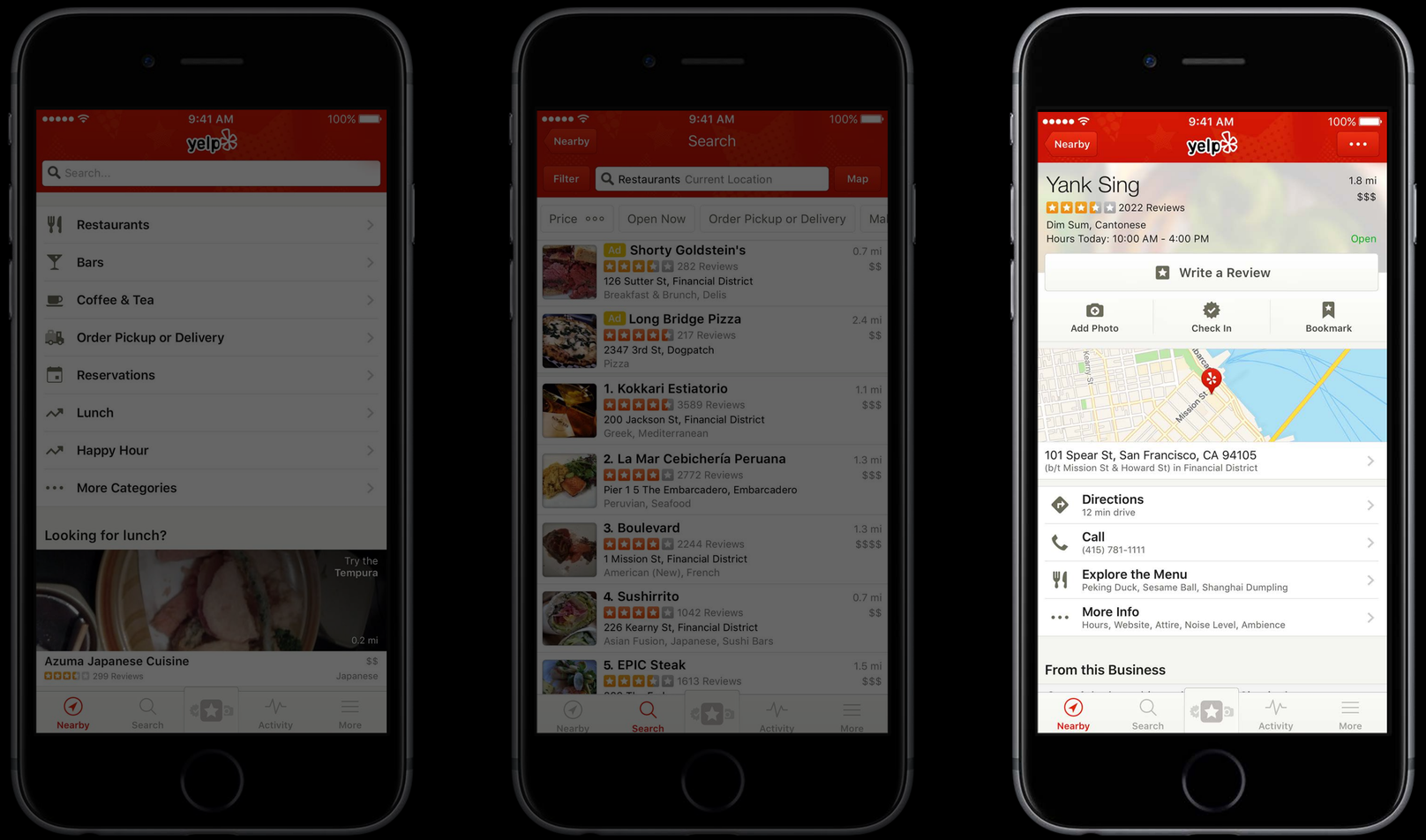
NSUserActivity



NSUserActivity

NSUserActivity

Capturing app state

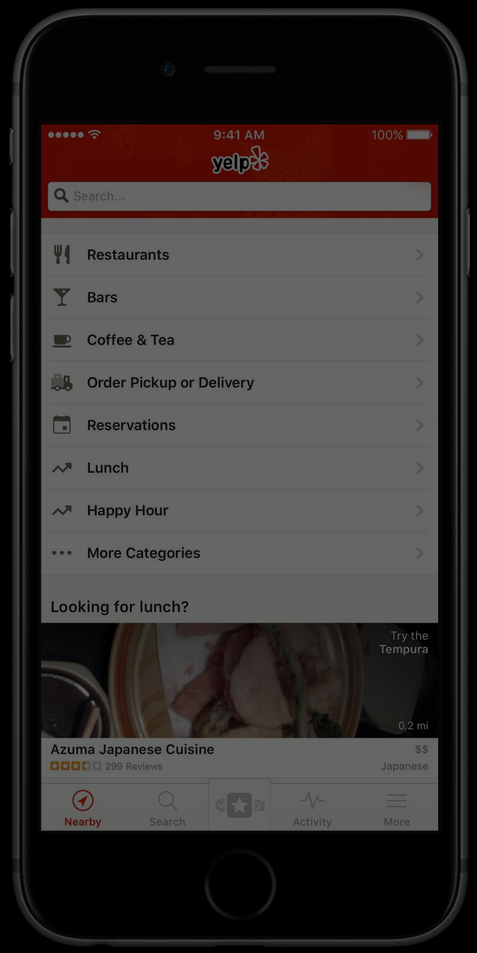


NSUserActivity

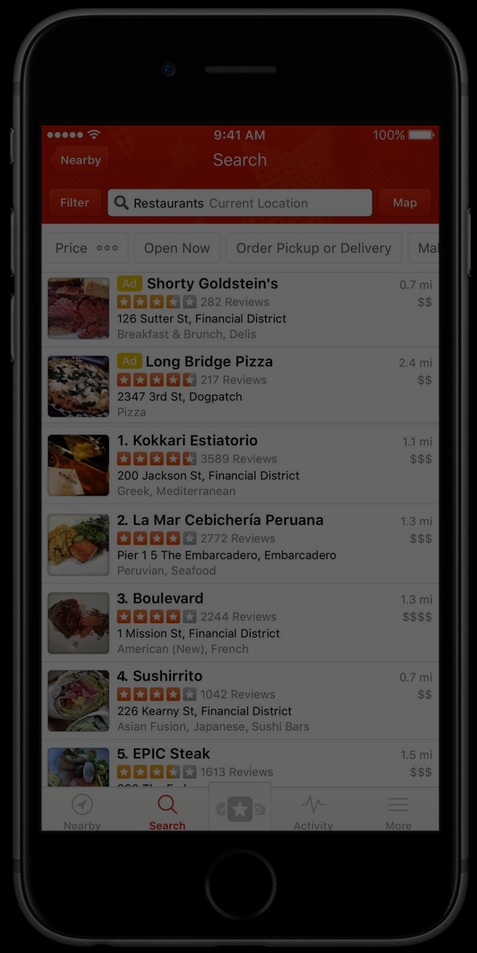
NSUserActivity

NSUserActivity

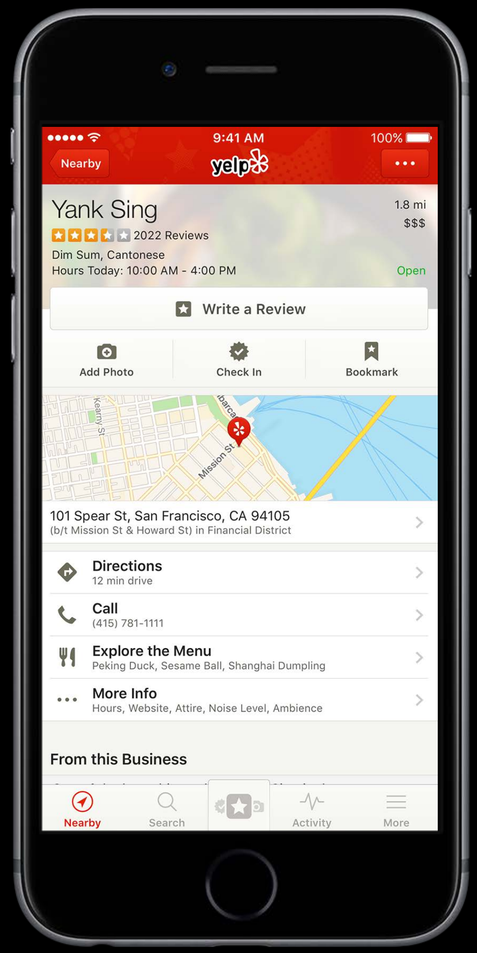
Capturing app state



NSUserActivity

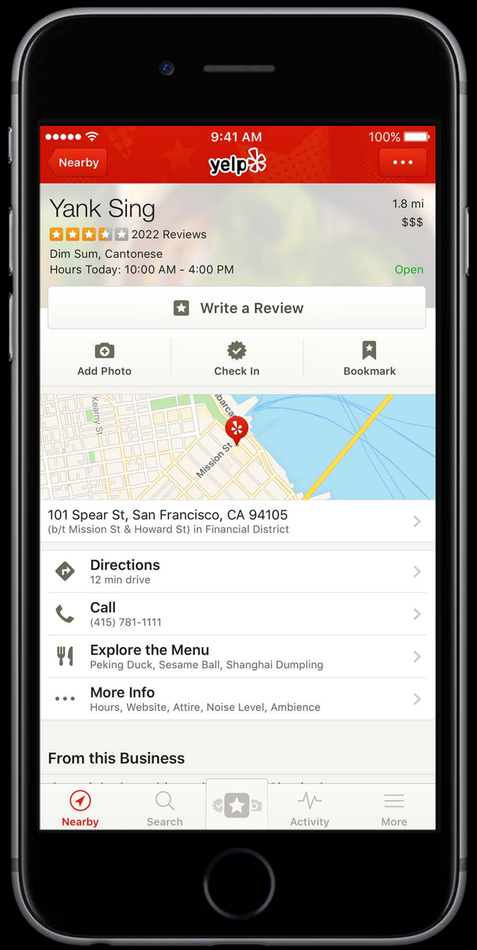


NSUserActivity

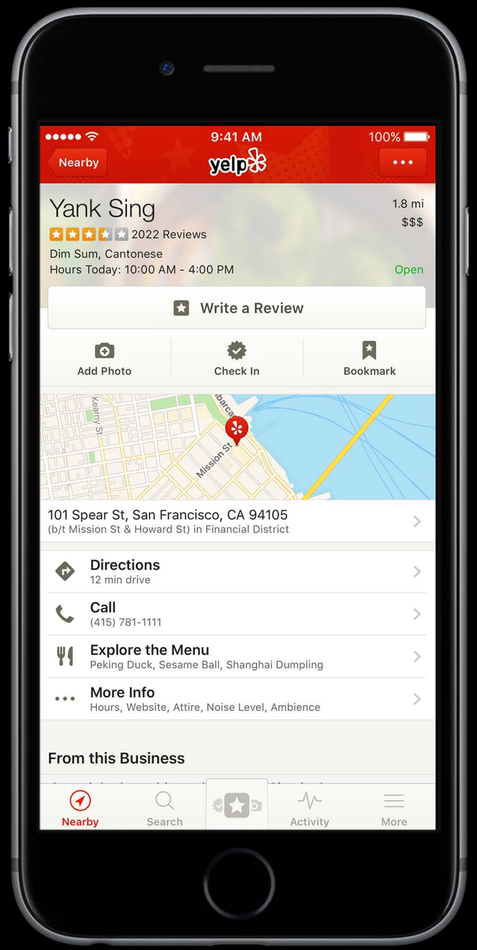


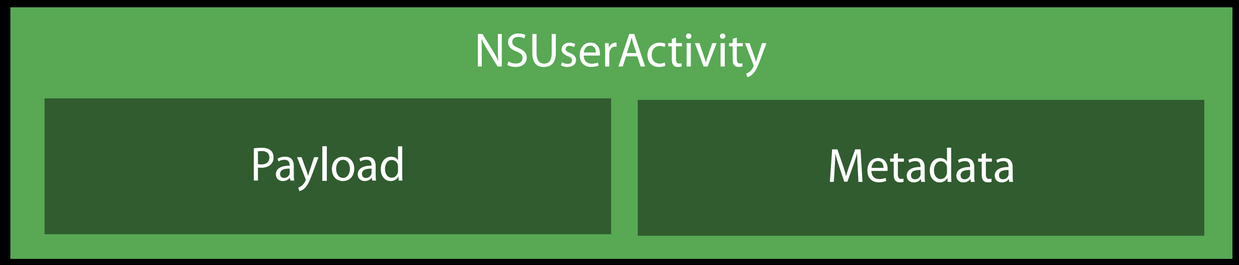
NSUserActivity

NSUserActivity

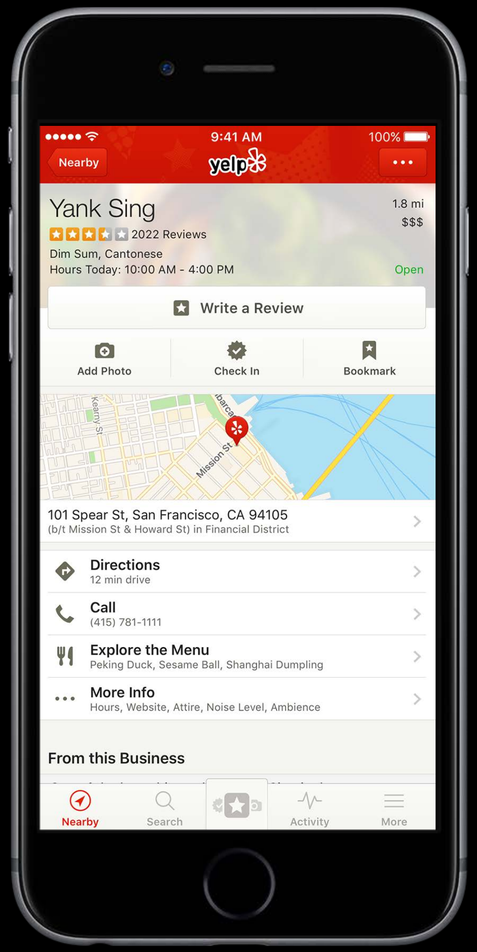


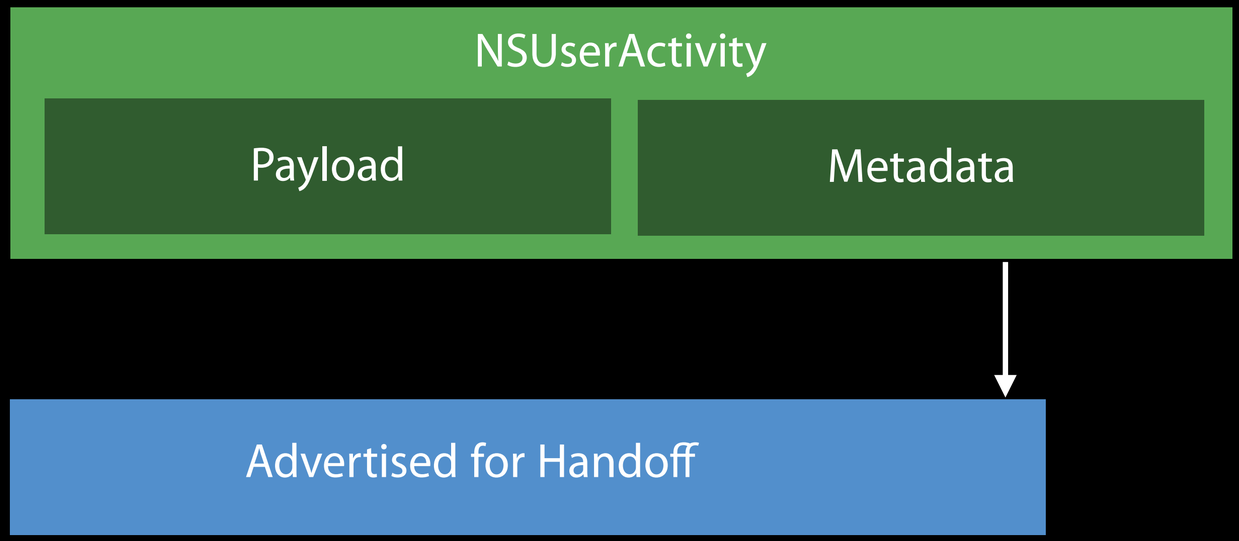
NSUserActivity



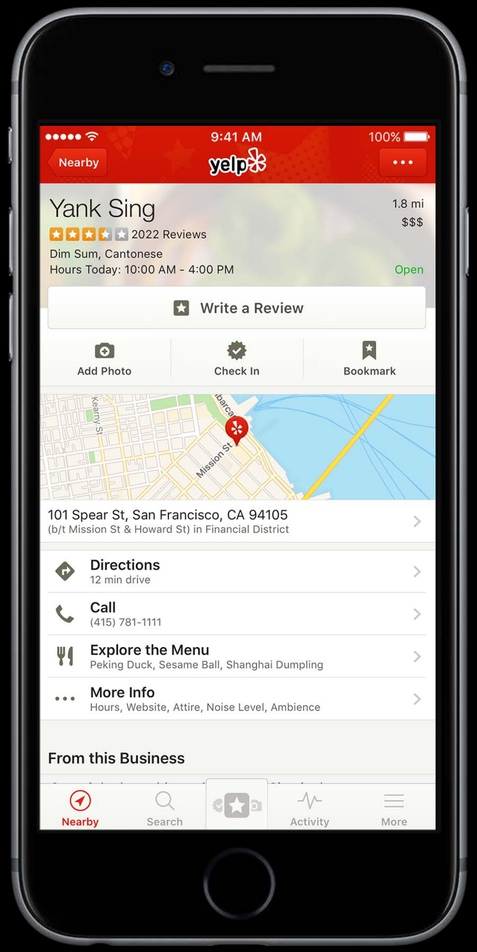


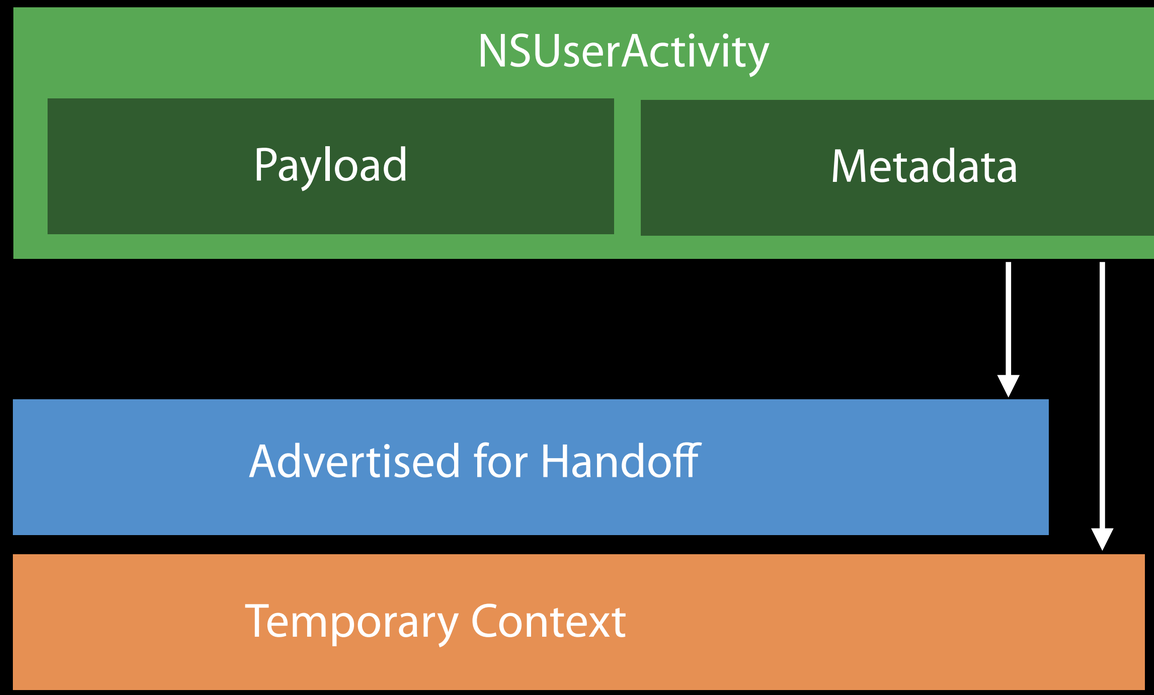
NSUserActivity



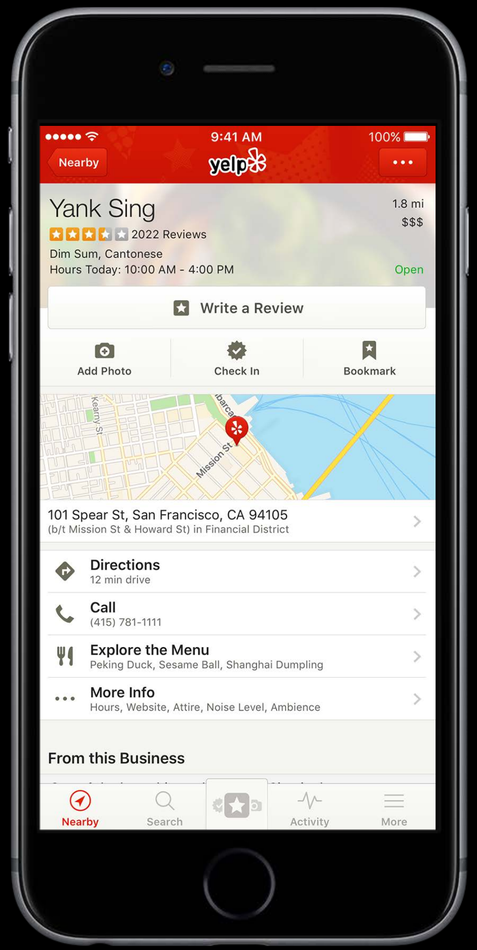


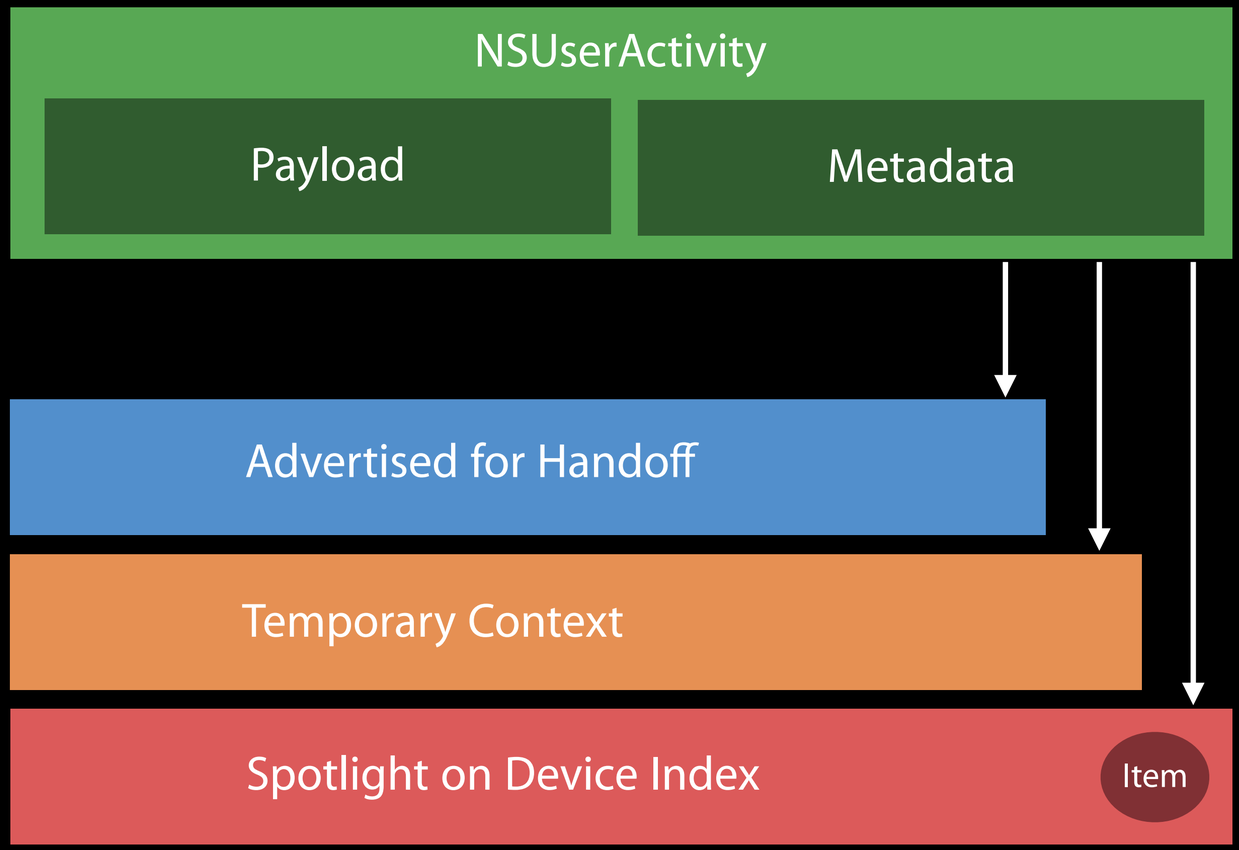
NSUserActivity





NSUserActivity





Related Sessions

Making the Most of Search APIs	Pacific Heights	Thursday 11:00AM
Adopting Handoff on iOS and OS X		WWDC 2014
Introducing Search APIs		WWDC 2015

Creating Activities

Creating Activities

Create user activity object

```
let activity = NSUserActivity(activityType: "com.example.view-location")
activity.title = "Yank Sing"
```

Creating Activities

Create user activity object

```
let activity = NSUserActivity(activityType: "com.example.view-location")
activity.title = "Yank Sing"
```

Enable capabilities

```
activity.isEligibleForHandoff = true
activity.isEligibleForSearch = true
activity.isEligibleForPublicIndexing = true
```

Creating Activities

Create user activity object

```
let activity = NSUserActivity(activityType: "com.example.view-location")
activity.title = "Yank Sing"
```

Enable capabilities

```
activity.isEligibleForHandoff = true
activity.isEligibleForSearch = true
activity.isEligibleForPublicIndexing = true
```

Specify userInfo

```
activity.userInfo = ["id": "yank-sing-san-francisco"]
```

Creating Activities

Provide metadata

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "com.example.location")
// Provide values for all relevant attributes
attributes.thumbnailURL = myThumbnailURL
activity.keywords = ["dim sum", "cantonese", "restaurant"]
activity.contentAttributeSet = attributes
```

Creating Activities

Provide metadata

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "com.example.location")
// Provide values for all relevant attributes
attributes.thumbnailURL = myThumbnailURL
activity.keywords = ["dim sum", "cantonese", "restaurant"]
activity.contentAttributeSet = attributes
```

Restoring on the web

```
activity.webpageURL = "http://www.example.com/yank-sing-san-francisco"
```

Creating Activities

Provide metadata

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "com.example.location")
// Provide values for all relevant attributes
attributes.thumbnailURL = myThumbnailURL
activity.keywords = ["dim sum", "cantonese", "restaurant"]
activity.contentAttributeSet = attributes
```

Restoring on the web

```
activity.webpageURL = "http://www.example.com/yank-sing-san-francisco"
```

```
activity.becomeCurrent()
```

Continuing Activities

UIApplicationDelegate

```
func application(UIApplication, continueUserActivity userActivity:
    NSUserActivity, restorationHandler: [AnyObject]? -> Void) -> Bool {

    if userActivity.activityType == "com.example.view-location" {

        // Restore state for userActivity and userInfo

    }

    return true
}
```

Continuing Activities

UIApplicationDelegate

```
func application(UIApplication, continueUserActivity userActivity:
    NSUserActivity, restorationHandler: [AnyObject]? -> Void) -> Bool {

    if userActivity.activityType == "com.example.view-location" {

        // Restore state for userActivity and userInfo

    }

    return true
}
```

Continuing Activities

UIApplicationDelegate

```
func application(UIApplication, continueUserActivity userActivity:
    NSUserActivity, restorationHandler: [AnyObject]? -> Void) -> Bool {

    if userActivity.activityType == "com.example.view-location" {

        // Restore state for userActivity and userInfo

    }

    return true
}
```

1. Handoff
2. Spotlight Search
3. Contextual Siri Reminders
4. Location Suggestions
5. Contextual Siri Requests
6. Contact Interactions

- ✓ Handoff
- ✓ Spotlight Search
- ✓ Contextual Siri Reminders
- 4 Location Suggestions
- 5 Contextual Siri Requests
- 6 Contact Interactions

Find a Place to Eat

1

Steps

Find a Place to Eat

Switch App

Text Friends

3

Steps

Find a Place to Eat

Switch App

Switch App

Copy Address

Text Friends

5

Steps

Find a Place to Eat

Switch App

Switch App

Paste Address

Text Friends

Copy Address

Switch App

7

Steps

Find a Place to Eat

Switch App

Switch App

Text Friends

Get Directions

Switch App

Copy Address

Switch App

Switch App

Paste Address

9

Steps

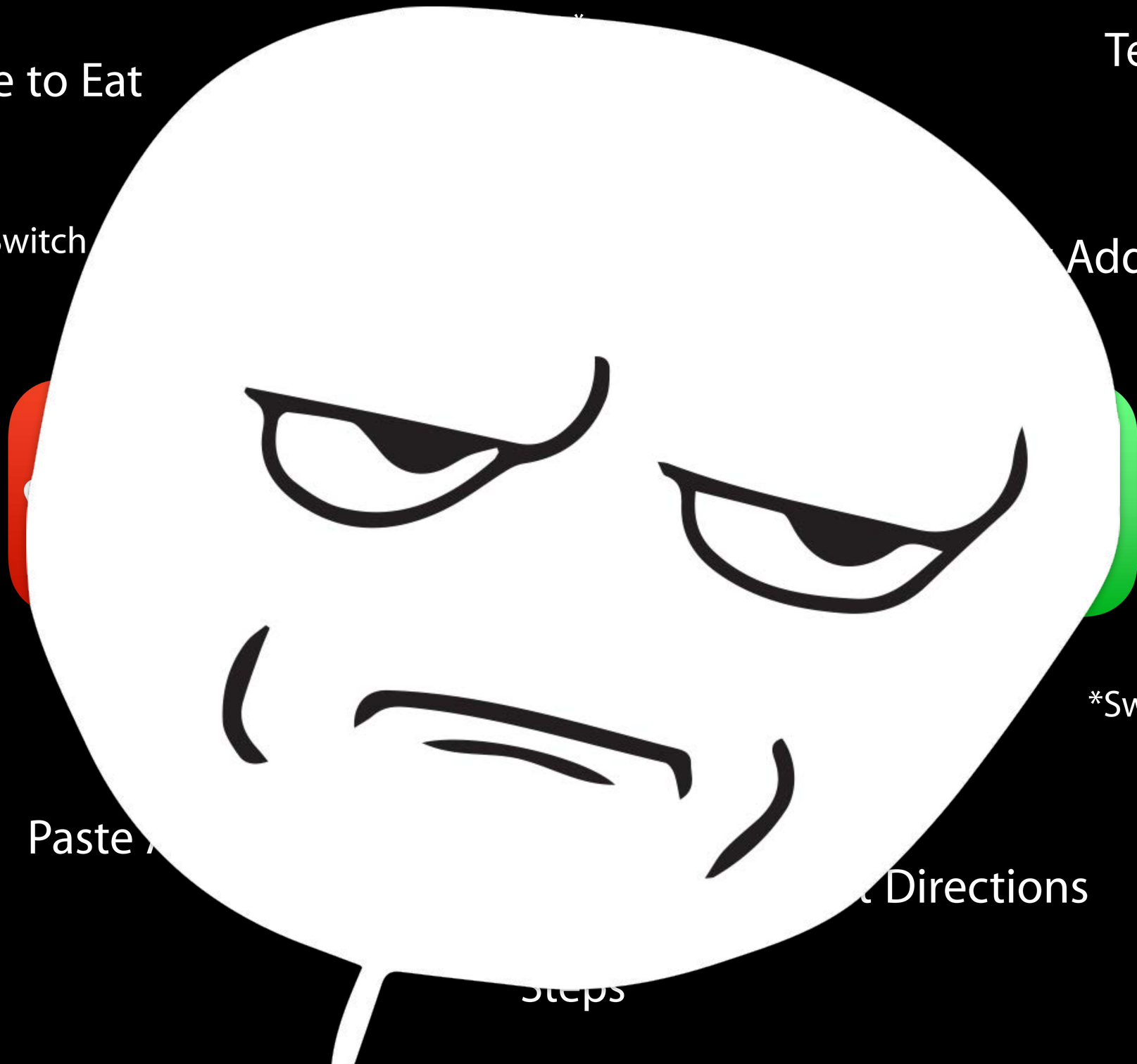
Get Directions

Find a Place to Eat

Text Friends

*Switch

Address



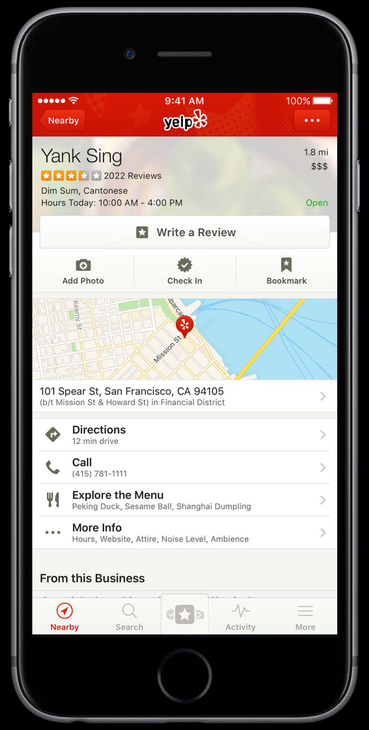
Switch App

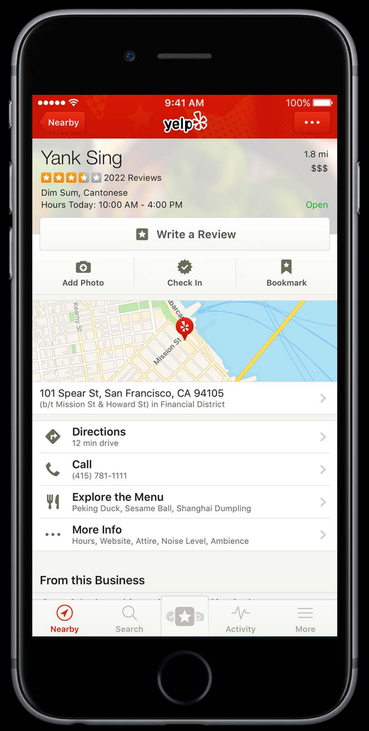
Switch App

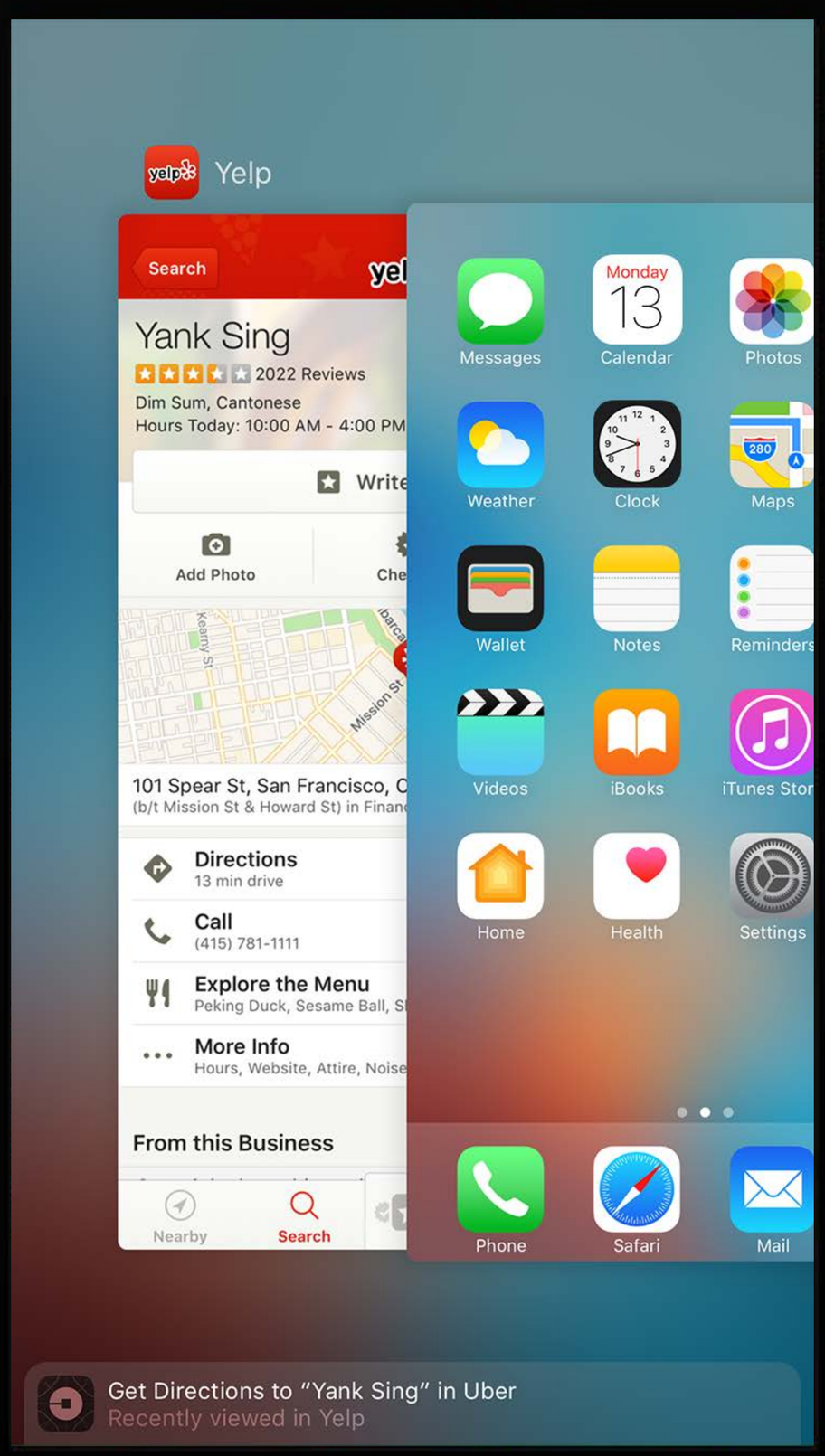
Paste

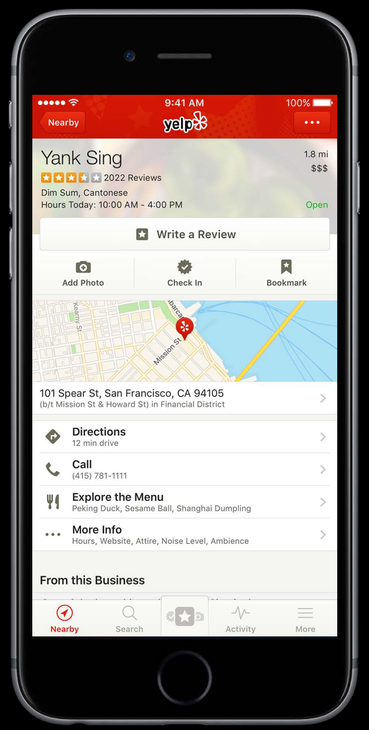
Directions

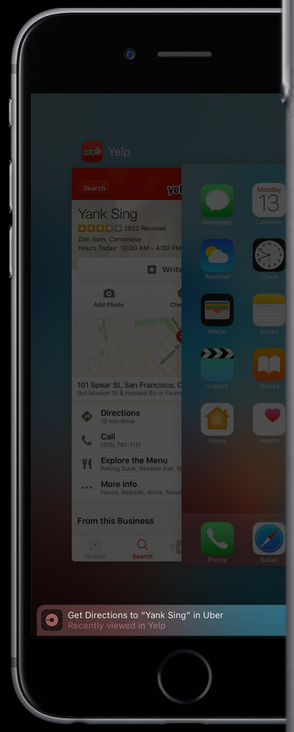
Steps


9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
Spear St
Mission St
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More

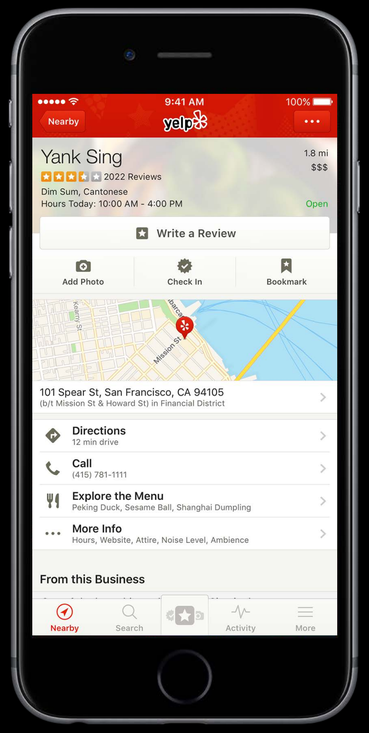

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More

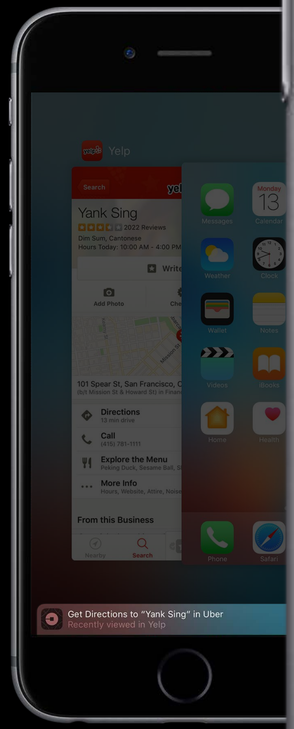

Yelp
Search
Yank Sing
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Add Photo
Directions
13 min drive
Call
(415) 781-1111
Explore the Menu
More Info
Hours, Website, Attire, Noise
From this Business
Nearby
Search
Messages
Calendar
Monday
13
Photos
Weather
Clock
Maps
Wallet
Notes
Reminders
Videos
iBooks
Home
Health
Settings
Phone
Safari
Mail
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

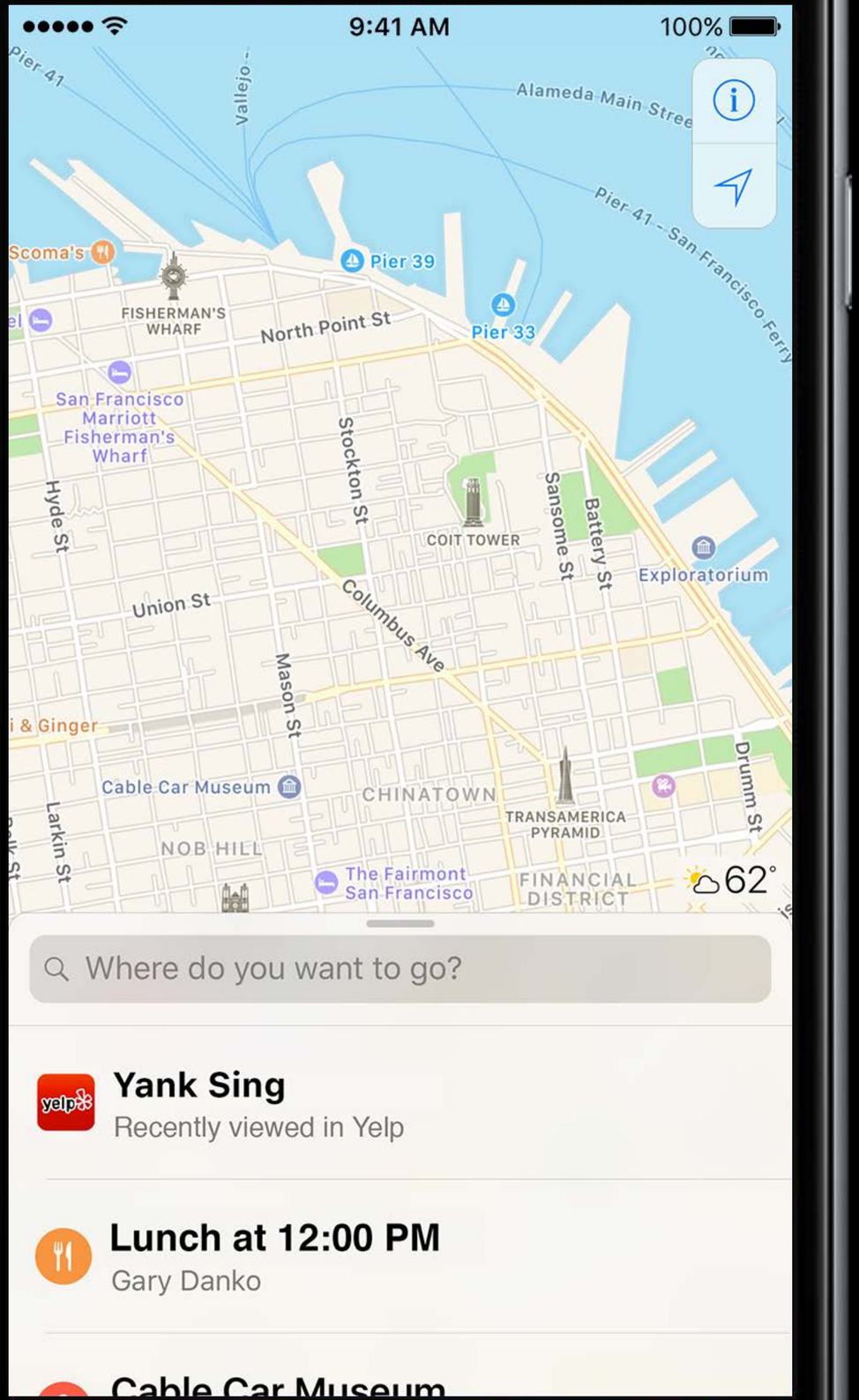

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More

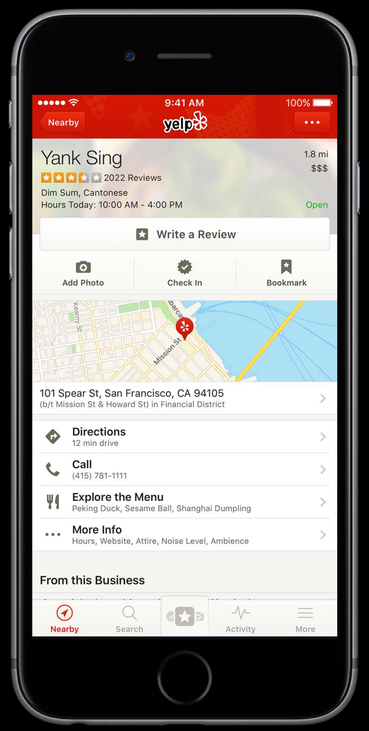

Yelp
Yank Sing
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Add Photo
101 Spear St, San Francisco, C
Directions
Call
(415) 781-1111
Explore the Menu
More Info
From this Business
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

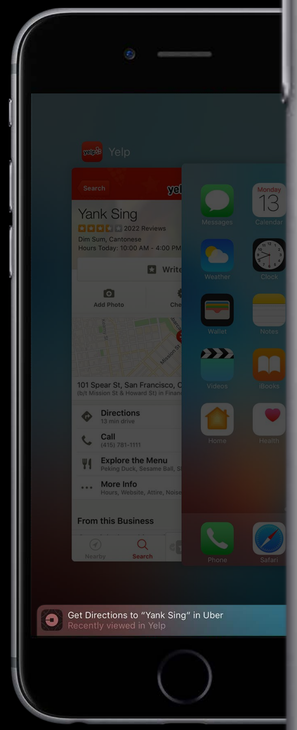

9:41 AM
100%
Sofiane
Hey
Here's a good one
How many developers does it take to change a light bulb?
None, that's a hardware problem.
Anyway, I found a place to eat...
Delivered
Let's meet at
Yank Sing from Yelp
101 Spear St San Francisco CA 94105
q w e r t y u i o p
a s d f g h j k l
z x c v b n m
123
space
return

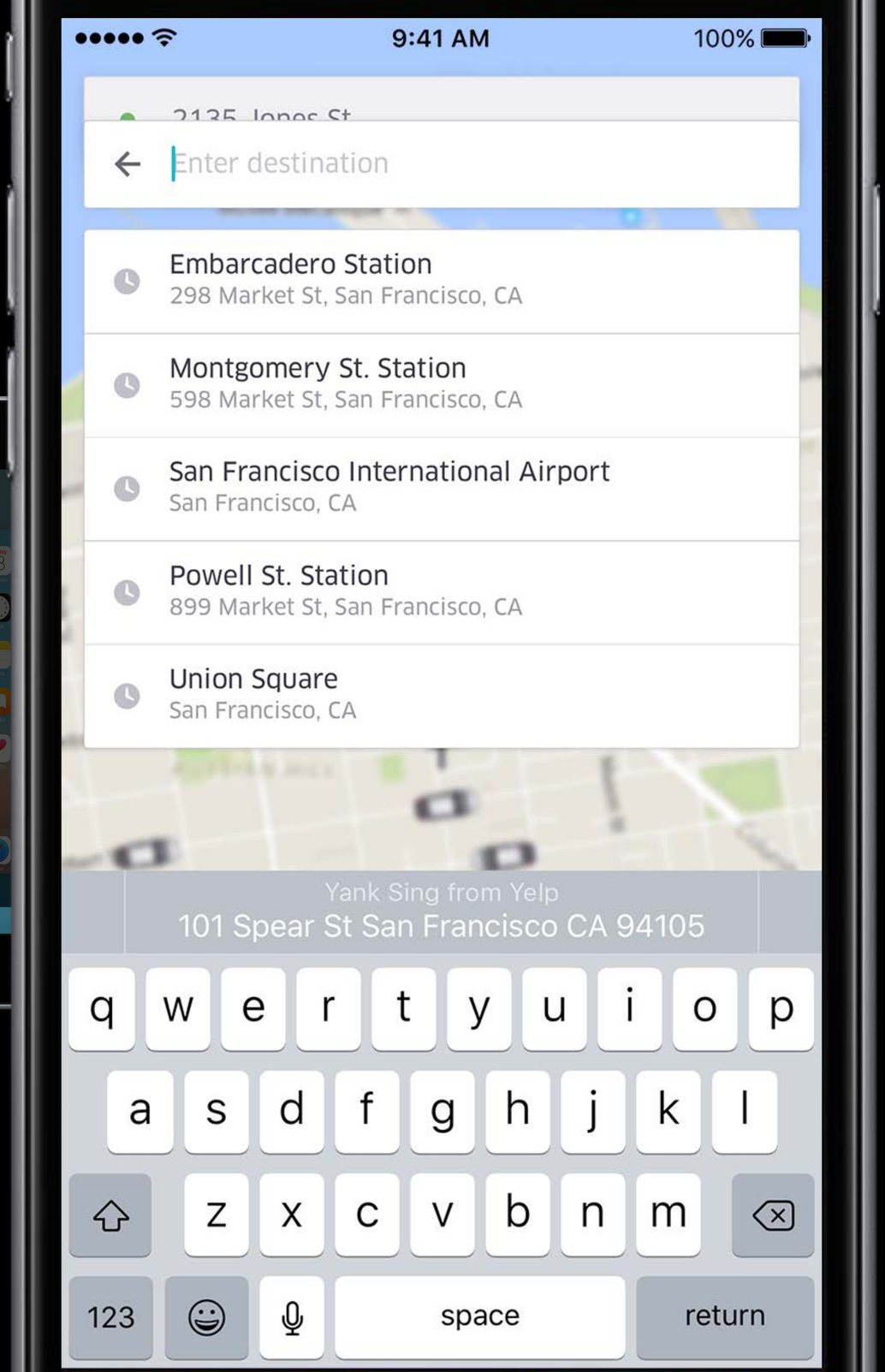

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More

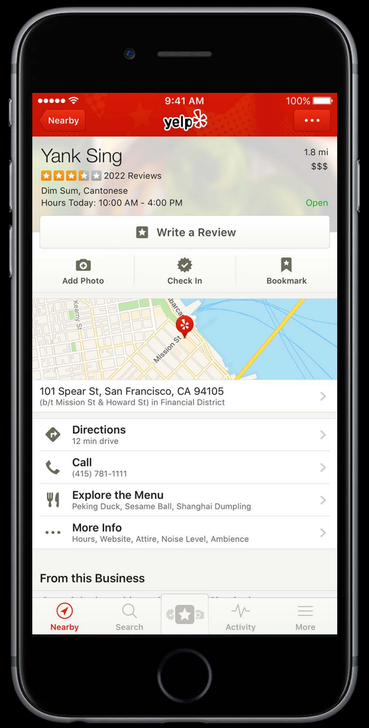

Yelp
Yank Sing
Directions
Call
Explore the Menu
More Info
From this Business
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

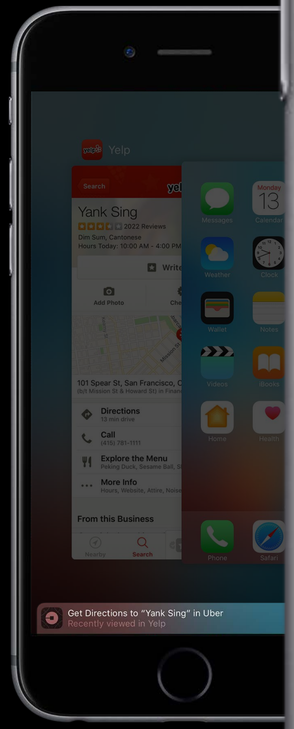

9:41 AM
100%
Pier 41
Vallejo
Alameda Main Stree
Pier 41 - San Francisco Ferry
Scoma's
Pier 39
FISHERMAN'S WHARF
North Point St
Pier 33
San Francisco Marriott Fisherman's Wharf
Stockton St
Hyde St
COIT TOWER
Sansome St
Battery St
Exploratorium
Union St
Columbus Ave
Mason St
Cable Car Museum
CHINATOWN
TRANSAMERICA PYRAMID
Drumm St
Larkin St
NOB HILL
The Fairmont San Francisco
FINANCIAL DISTRICT
62°
Where do you want to go?
Yank Sing
Recently viewed in Yelp
Lunch at 12:00 PM
Gary Danko
Cable Car Museum

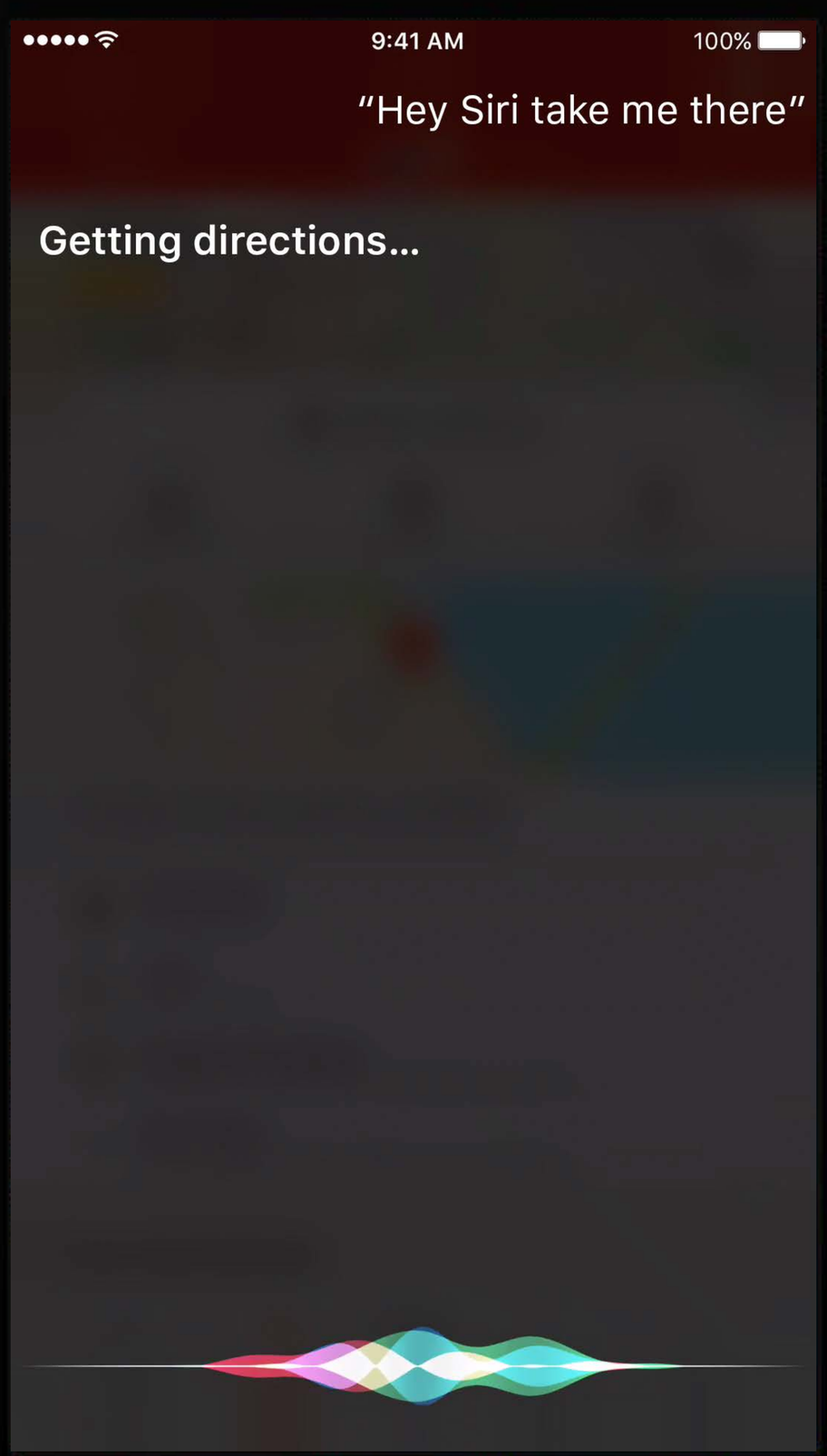

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More

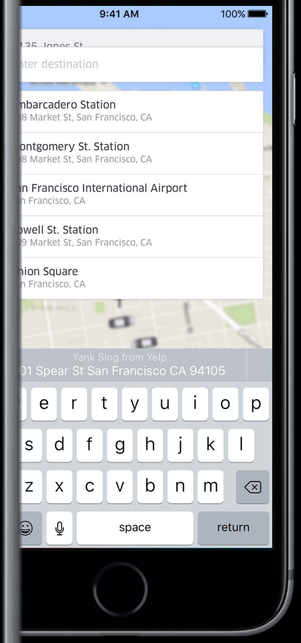

Yelp
Yank Sing
Directions
Call
Explore the Menu
More Info
From this Business
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

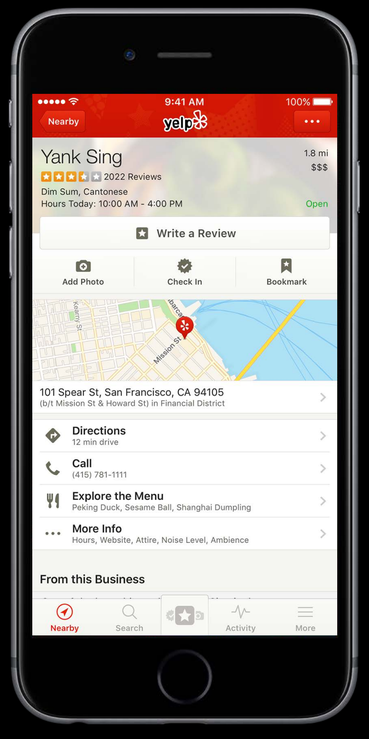

9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA
Yank Sing from Yelp
101 Spear St San Francisco CA 94105
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

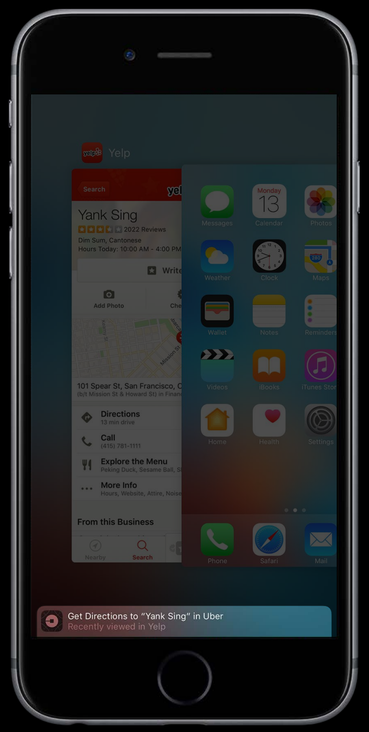

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More


Yelp
Yank Sing
Directions
Call
Explore the Menu
More Info
From this Business
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

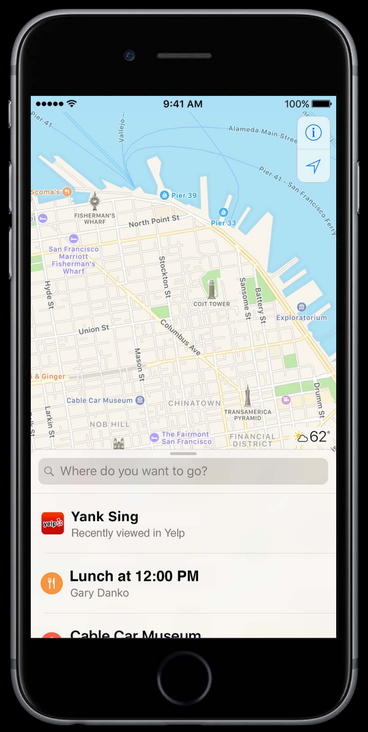

9:41 AM
100%
"Hey Siri take me there"
Getting directions...

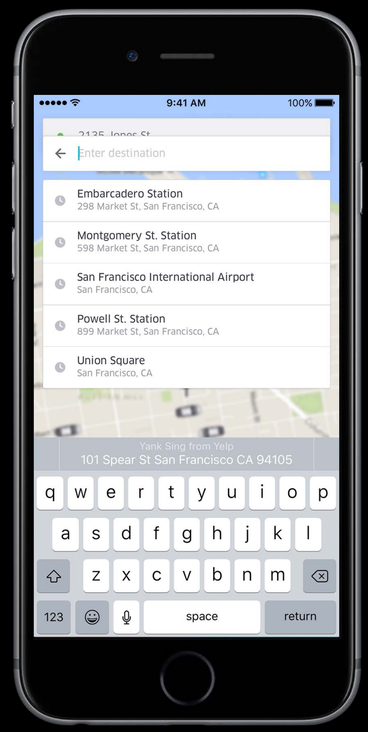

9:41 AM
100%
Enter destination
Embarcadero Station
San Francisco International Airport
Powell St. Station
Union Square
Yank Sing from Yelp
space
return

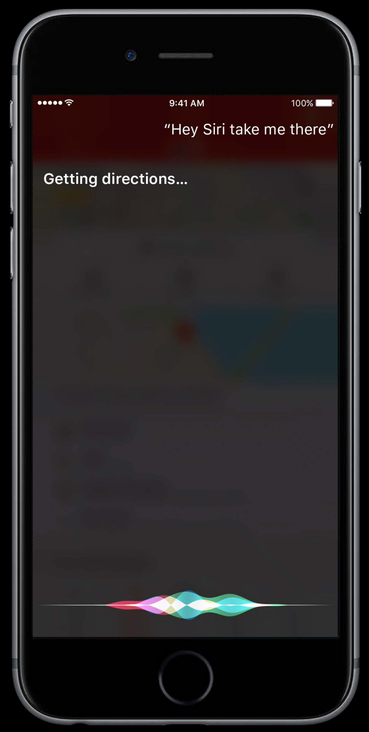

9:41 AM
100%
Nearby
yelp
Yank Sing
1.8 mi
$$$
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Open
Write a Review
Add Photo
Check In
Bookmark
101 Spear St, San Francisco, CA 94105
(b/t Mission St & Howard St) in Financial District
Directions
12 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, Shanghai Dumpling
More Info
Hours, Website, Attire, Noise Level, Ambience
From this Business
Nearby
Search
Activity
More


Yelp
Yank Sing
Directions
Call
Explore the Menu
More Info
From this Business
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp


9:41 AM
100%
Sofiane
Hey
Here's a good one
How many developers does it take to change a light bulb?
None, that's a hardware problem.
Anyway, I found a place to eat...
Delivered
Let's meet at
Yank Sing from Yelp
101 Spear St San Francisco CA 94105


9:41 AM
100%
Where do you want to go?
Yank Sing
Recently viewed in Yelp
Lunch at 12:00 PM
Gary Danko


9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA
Yank Sing from Yelp
101 Spear St San Francisco CA 94105


9:41 AM
100%
"Hey Siri take me there"
Getting directions...

From Yelp

From Your App

Location Suggestions

Background

NEW

Capture locations viewed by the user

MapKit and CoreSpotlight APIs

Integrate with Siri, Maps, Keyboards, Multitasking, and more

Location Suggestions

MapKit-based apps

NEW

Reuse the same user activity

```
let activity = NSUserActivity(activityType: "com.example.view-location")
```

Location Suggestions

MapKit-based apps

NEW

Reuse the same user activity

```
let activity = NSUserActivity(activityType: "com.example.view-location")
```

Set MKMapItem

```
activity.mapItem = myMapItem
```

Location Suggestions

Apps adopting app search

NEW

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

Location Suggestions

Apps adopting app search

NEW

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

Location Suggestions

Apps adopting app search

NEW

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

Location Suggestions

Apps adopting app search

NEW

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

Location Suggestions

Apps adopting app search

NEW

```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

Location Suggestions

Apps adopting app search

NEW

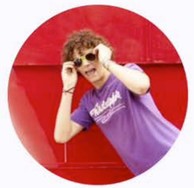
```
let attributes = CSSearchableItemAttributeSet(itemContentType: "location")
attributes.namedLocation = "Apple Inc."
attributes.subThoroughfare = "1"
attributes.thoroughfare = "Infinite Loop"
attributes.city = "Cupertino"
attributes.stateOrProvince = "CA"
attributes.country = "United States”
attributes.latitude = 37.33072
attributes.longitude = -122.029674
attributes.phoneNumbers = ["(800) 275-2273"]
attributes.supportsPhoneCall = true
attributes.supportsNavigation = true
```

- ✓ Handoff
- ✓ Spotlight Search
- ✓ Contextual Siri Reminders
- 4 Location Suggestions
- 5 Contextual Siri Requests
- 6 Contact interactions

- ✓ Handoff
- ✓ Spotlight Search
- ✓ Contextual Siri Reminders
- ✓ Location Suggestions
- ✓ Contextual Siri Requests
- 6 Contact Interactions

9:41 AM 100%

Contacts Edit



John Appleseed

message call video mail

iPhone ★
(408) 555-0621

work
j.appleseed@icloud.com

WhatsApp
j.appleseed

Notes

Send Message

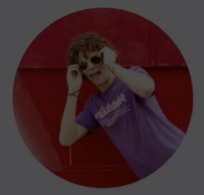
Share Contact

Add to Favorites

Share My Location

9:41 AM 100%

Contacts Edit



John Appleseed

message call video mail

iPhone ★
(408) 555-0621

work
j.appleseed@icloud.com

WhatsApp
j.appleseed

Notes

Send Message

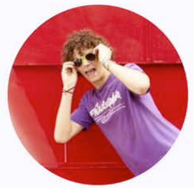
Share Contact

Add to Favorites

Share My Location

9:41 AM 100%

Contacts Edit



John Appleseed

message call video mail

iPhone ★
(408) 555-0621

work
j.appleseed@icloud.com

WhatsApp
j.appleseed

Notes

Send Message

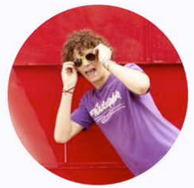
Share Contact

Add to Favorites

Share My Location

9:41 AM 100%

Contacts Edit



John Appleseed

message call video mail

iPhone ★
(408) 555-0621

work
j.appleseed@icloud.com

WhatsApp
j.appleseed

Notes

Send Message

Share Contact

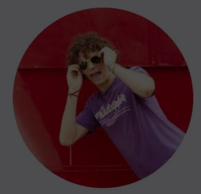
Add to Favorites

Share My Location


9:41 AM
100%
Contacts
Edit
John Appleseed
WhatsApp
call
video
mail
iPhone
(408) 555-0621
work
j.appleseed@icloud.com
WhatsApp
j.appleseed
Notes
Send Message
Share Contact
Add to Favorites
Favorites
Recents
Contacts
Keypad
Voicemail

9:41 AM 100%

Contacts Edit



John Appleseed

WhatsApp

call video mail

iPhone
(408) 555-0621

work
j.appleseed@icloud.com

WhatsApp
j.appleseed

Notes

Send Message

Share Contact

Add to Favorites

Favorites Recents Contacts Keypad Voicemail

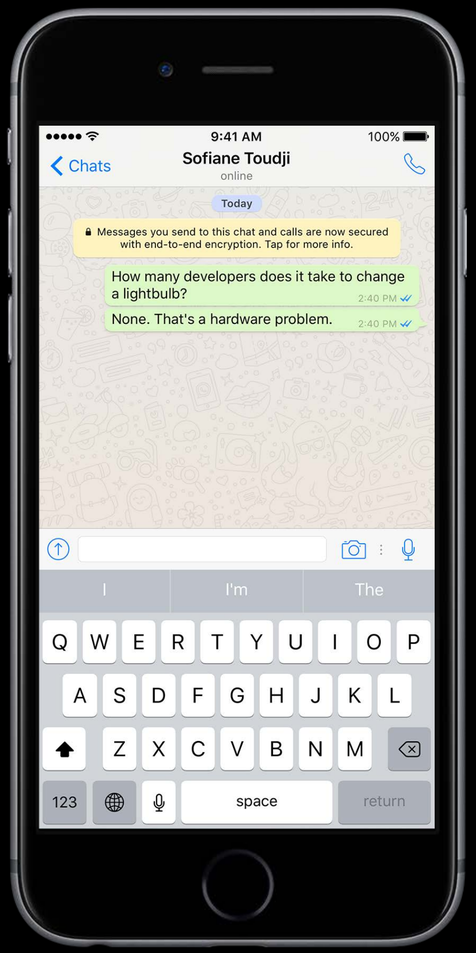
NSUserActivity + Intents

Introducing SiriKit	Presidio	Wednesday 5:00PM
Extending Your Apps with SiriKit	Presidio	Thursday 1:40PM

Contact Interactions

Donating interactions

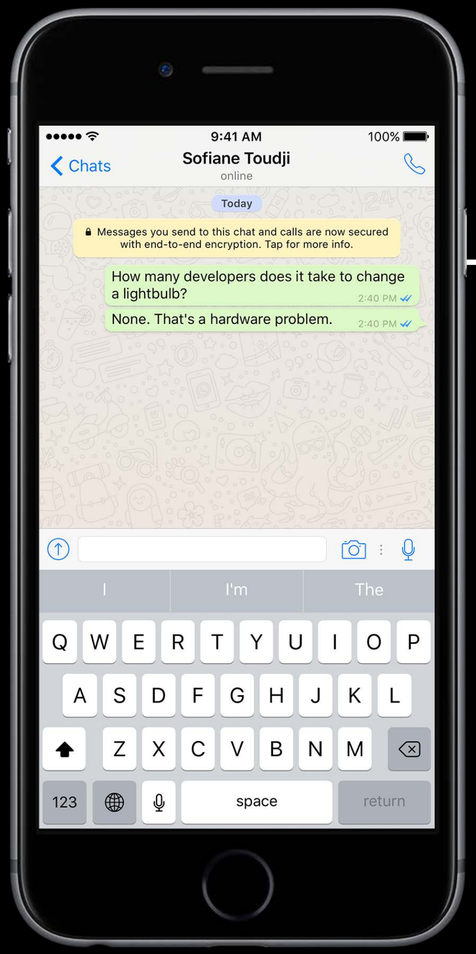
NEW



Contact Interactions

Donating interactions

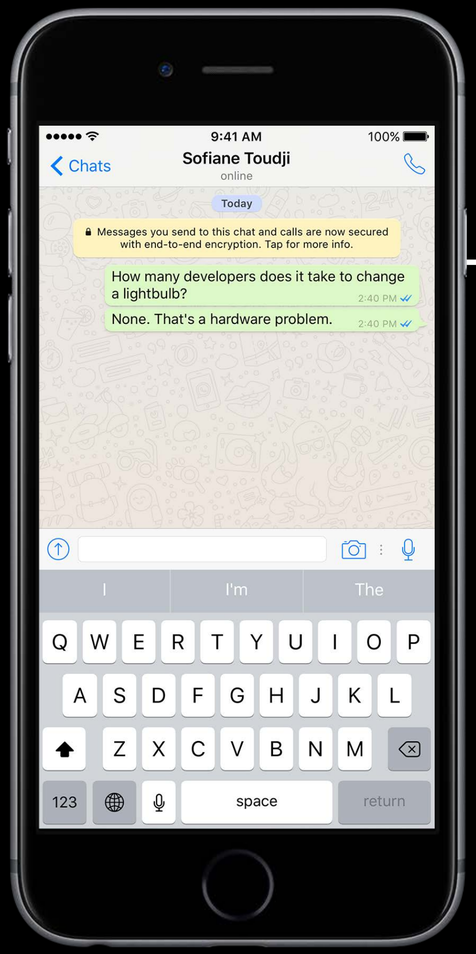
NEW



Contact Interactions

Donating interactions

NEW

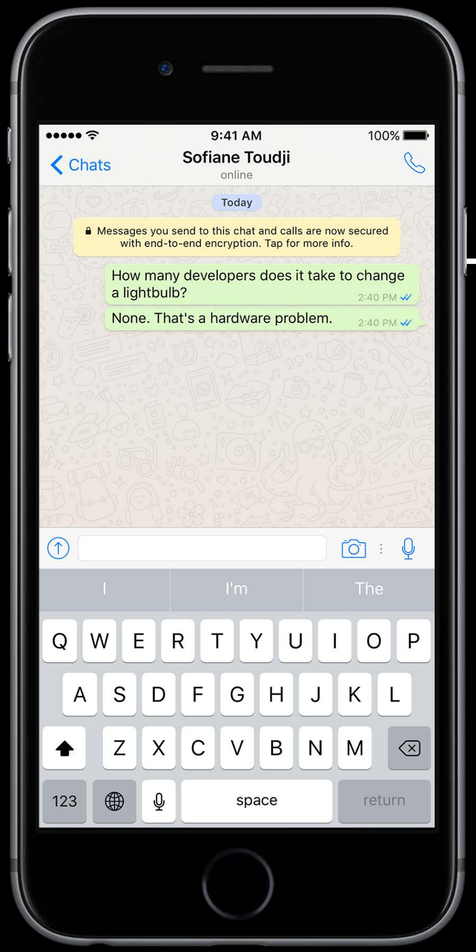


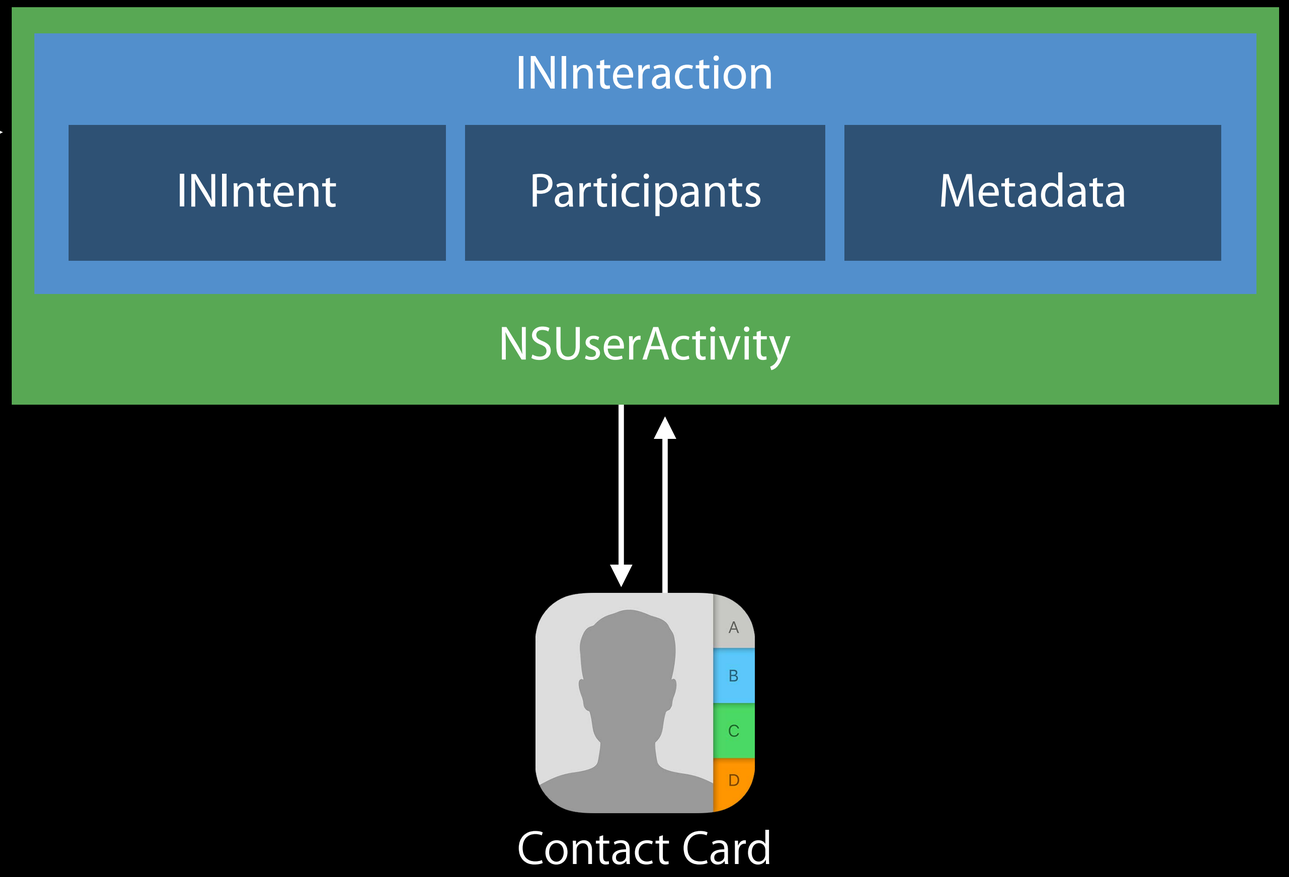
Contact Card

Contact Interactions

Donating interactions

NEW





Contact Interactions

NEW

Donating interactions

Specify sender and recipient

```
let sender = INPerson(handle: "j.appleseed@icloud.com",
                      displayName: "John Appleseed",
                      contactIdentifier: nil)

let recipient = INPerson(handle: "janedaniels@icloud.com",
                         displayName: "Jane Daniels",
                         contactIdentifier: nil)
```

Contact Interactions

NEW

Donating interactions

Create an intent

```
let intent = INSendMessageIntent(recipients: [recipient],
                                 content: nil
                                 serviceName: INMessageServiceNameWhatsApp
                                 sender: sender)
```

Contact Interactions

Donating interactions

NEW

Create an intent

```
let intent = INSendMessageIntent(recipients: [recipient],
                                 content: nil
                                 serviceName: INMessageServiceNameWhatsApp
                                 sender: sender)
```

Contact Interactions

Donating interactions

NEW

Create an intent

```
let intent = INSendMessageIntent(recipients: [recipient],
                                 content: nil
                                 serviceName: INMessageServiceNameWhatsApp
                                 sender: sender)
```

Three communication intents

- INSendMessageIntent
- INStartAudioCallIntent
- INStartVideoCallIntent

Contact Interactions

Donating interactions

NEW

Create an intent

```
let intent = INSendMessageIntent(recipients: [recipient],
                                 content: nil
                                 serviceName: INMessageServiceNameWhatsApp
                                 sender: sender)
```

Three communication intents

- INSendMessageIntent
- INStartAudioCallIntent
- INStartVideoCallIntent

Contact Interactions

Donating interactions

NEW

Create an interaction

```
let interaction = INInteraction(intent: intent,
                                response: INIntent.responseSuccess)

interaction.direction = INInteractionDirectionOutgoing
interaction.intentHandlingStatus = INIntentHandlingStatusDone
```

Contact Interactions

Donating interactions

NEW

Create an interaction

```
let interaction = INInteraction(intent: intent,
                                response: INIntent.responseSuccess)

interaction.direction = INInteractionDirectionOutgoing
interaction.intentHandlingStatus = INIntentHandlingStatusDone
```

Contact Interactions

Donating interactions

NEW

Create an interaction

```
let interaction = INInteraction(intent: intent,
                                response: INIntent.responseSuccess)

interaction.direction = INInteractionDirectionOutgoing
interaction.intentHandlingStatus = INIntentHandlingStatusDone
```

Donate interaction

```
interaction.donateInteraction(completion: { (error) in
    // Interaction donated
})
```

Handling Communication

UIApplicationDelegate

```
func application(UIApplication, continueUserActivity userActivity:
    NSUserActivity, restorationHandler: [AnyObject]? -> Void) -> Bool {

      if let intent = userActivity.interaction?.intent as? INSendMessageIntent {
          // Start communication with recipient
    }

    return true
}
```

Handling Communication

UIApplicationDelegate

```
func application(UIApplication, continueUserActivity userActivity:
    NSUserActivity, restorationHandler: [AnyObject]? -> Void) -> Bool {

        if let intent = userActivity.interaction?.intent as? INSendMessageIntent {
            // Start communication with recipient
    }

    return true
}
```

- ✓ Handoff
- ✓ Spotlight Search
- ✓ Contextual Siri Reminders
- ✓ Location Suggestions
- ✓ Contextual Siri Requests
- 6 Contact Interactions

- Handoff
- Spotlight Search
- Contextual Siri Reminders
- Location Suggestions
- Contextual Siri Requests
- Contact Interactions

Best Practices

Best Practices

Lazy payload update

Use needsSave for lazy payload updates

```
func updateActivityForEachKeystroke() {
   activity.userInfo = ["textContent": textfield.text]
}
```

Best Practices

Lazy payload update

Use needsSave for lazy payload updates

```
func updateActivityForEachKeystroke() {
    activity.userInfo = ["textContent": textfield.text]
}
```

Best Practices

Lazy payload update

Use needsSave for lazy payload updates

```
func updateActivityForEachKeystroke() {
   activity.userInfo = ["textContent": textfield.text]
}
```

```
func updateActivityForEachKeystroke() {
   activity.needsSave = true
}


override func updateUserActivityState(_ activity: NSUserActivity) {
   activity.addUserInfoEntries(from: ["textContent": textfield.text])
}
```

Best Practices

Scope of current activity

Keep a strong reference to the current activity

```
func updateActivity() {
    let activity = NSUserActivity(activityType: "com.example.my-activity")
    // setup user activity
    activity.becomeCurrent()
}

// activity is released
```

Best Practices

Scope of current activity

Keep a strong reference to the current activity

```
func updateActivity() {
    let activity = NSUserActivity(activityType: "com.example.my-activity")
    // setup user activity
    activity.becomeCurrent()
}

// activity is released
```

Best Practices

Scope of current activity

Keep a strong reference to the current activity

```
func updateActivity() {
    let activity = NSUserActivity(activityType: "com.example.my-activity")
    // setup user activity
    self.userActivity = activity
    activity.becomeCurrent()
}
```

Best Practices

Scope of current activity

Keep a strong reference to the current activity

```
func updateActivity() {
    let activity = NSUserActivity(activityType: "com.example.my-activity")
    // setup user activity
    self.userActivity = activity
    activity.becomeCurrent()
}
```

Best Practices

Transfer a small userInfo payload

Transfer a small payload in the userInfo dictionary

```
let imageData: Data // Downloaded image data from a web service
let photo = UIImage(data: imageData)

activity.userInfo = ["photoData": imageData]
```

Best Practices

Transfer a small userInfo payload

Transfer a small payload in the userInfo dictionary

```
let imageData: Data // Downloaded image data from a web service
let photo = UIImage(data: imageData)

activity.userInfo = ["photoData": imageData]
```

Best Practices

Transfer a small userInfo payload

Transfer a small payload in the userInfo dictionary

```
let imageData: Data // Downloaded image data from a web service
let photo = UIImage(data: imageData)

activity.userInfo = ["photoData": imageData]
```

```
activity.userInfo = ["photoRemoteURL": imageRemoteURL]
```

Best Practices

Unique activity types

Use reverse-DNS notation for your activity types

```
let activity = NSUserActivity(activityType: "my-activity")
```

Best Practices

Unique activity types

Use reverse-DNS notation for your activity types

```
let activity = NSUserActivity(activityType: "my-activity")
```

Best Practices

Unique activity types

Use reverse-DNS notation for your activity types

```
let activity = NSUserActivity(activityType: "my-activity")
```

```
let activity = NSUserActivity(activityType: "com.example.view-location")
let activity = NSUserActivity(activityType: "com.example.search-location")
```

Demo

Proactive Toolbox app

NSUserActivity

Native apps

schema.org

Web apps

9:41 AM
yelp.com
100%

yelp

e.g. tacos, Mel's

Yank Sing

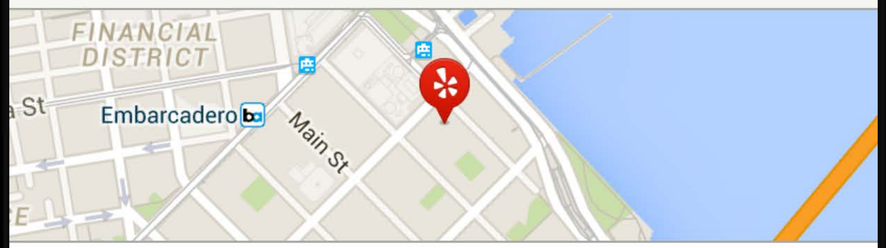
2022 Reviews

Dim Sum, Cantonese

Hours today: 10:00 am - 4:00 pm **Open**

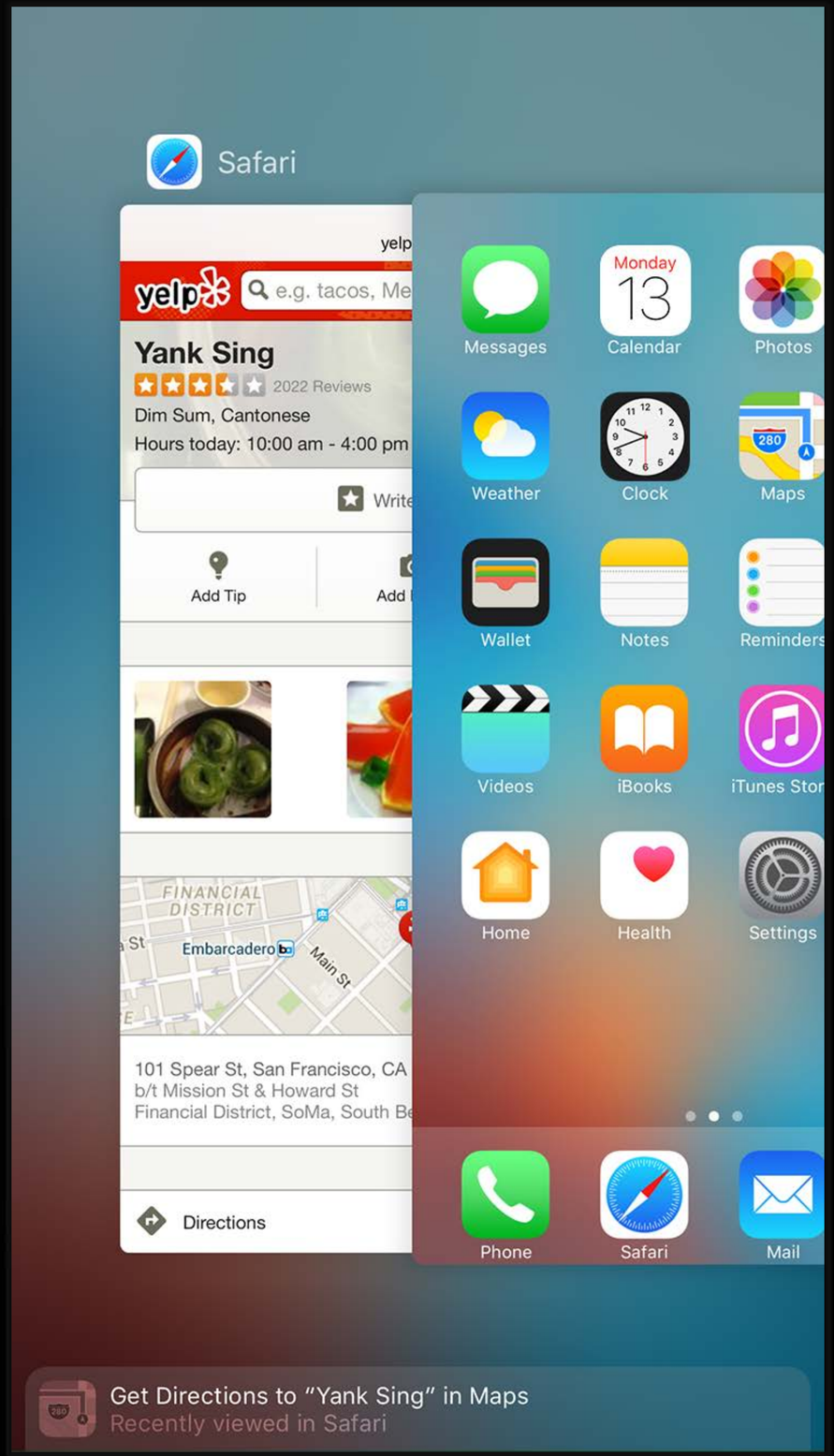
Write a Review

Add Tip | Add Photo | Bookmark



101 Spear St, San Francisco, CA 94105,
b/t Mission St & Howard St
Financial District, SoMa, South Beach

Directions


Safari
yelp
yelp
e.g. tacos, Me
Yank Sing
2022 Reviews
Dim Sum, Cantonese
Hours today: 10:00 am - 4:00 pm
Write
Add Tip
Add
FINANCIAL DISTRICT
Embarcadero
Main St
101 Spear St, San Francisco, CA
b/t Mission St & Howard St
Financial District, SoMa, South Be
Directions
Messages
Monday
13
Calendar
Photos
Weather
Clock
Maps
Wallet
Notes
Reminders
Videos
iBooks
iTunes Stor
Home
Health
Settings
Phone
Safari
Mail
Get Directions to “Yank Sing” in Maps
Recently viewed in Safari

schema.org

Background

Open web markup vocabulary standard

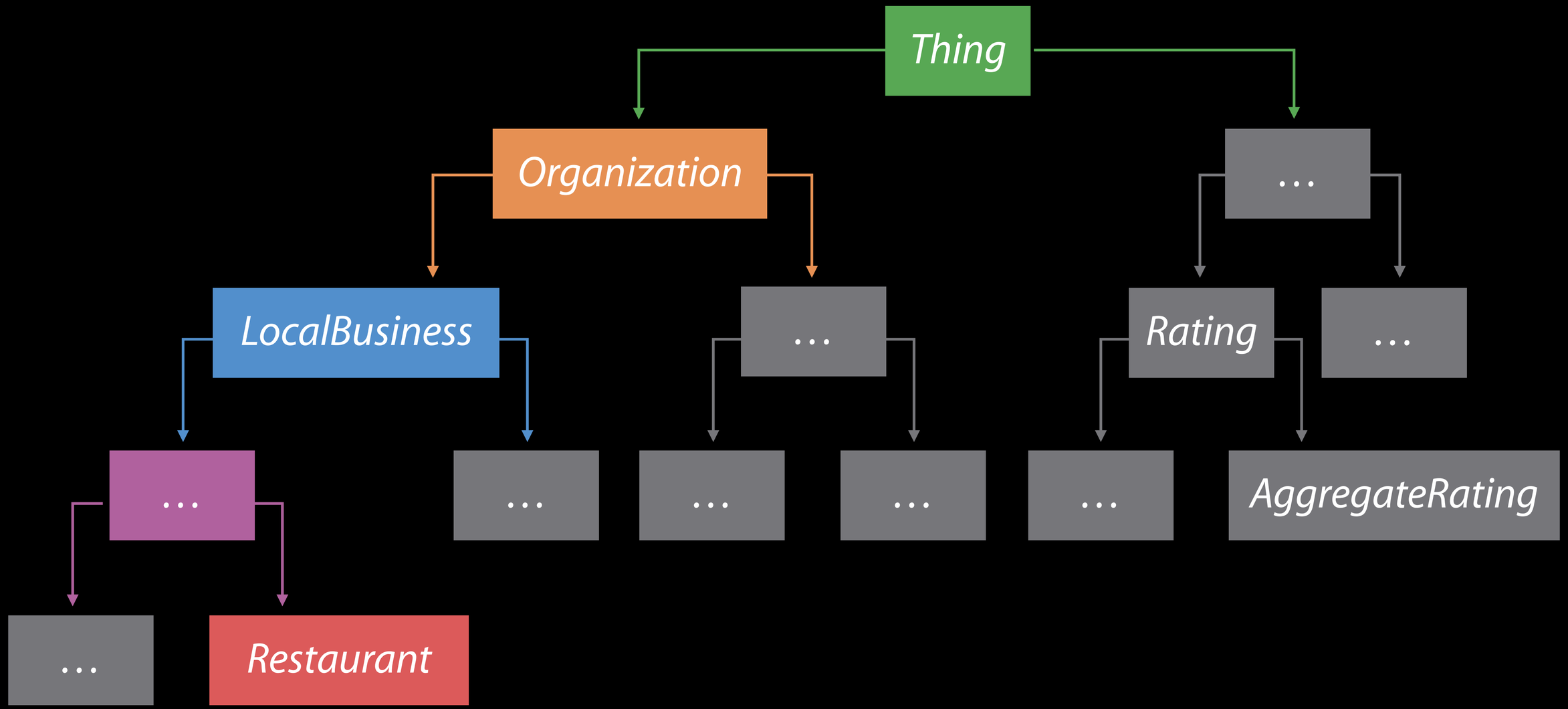
Structured metadata on web pages

500+ schemas representing various concepts

Some of the benefits of NSUserActivity for the web

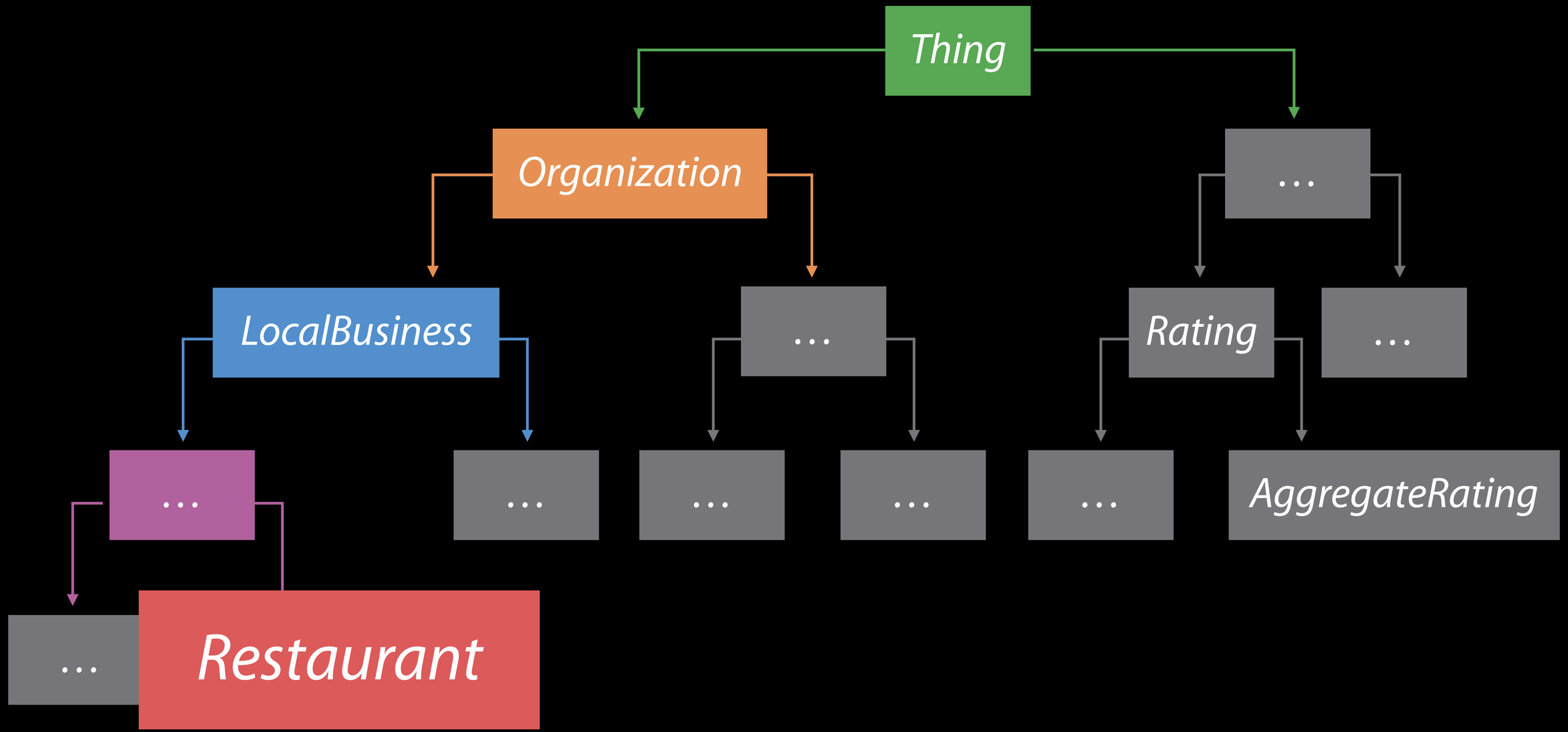
schema.org

Structure



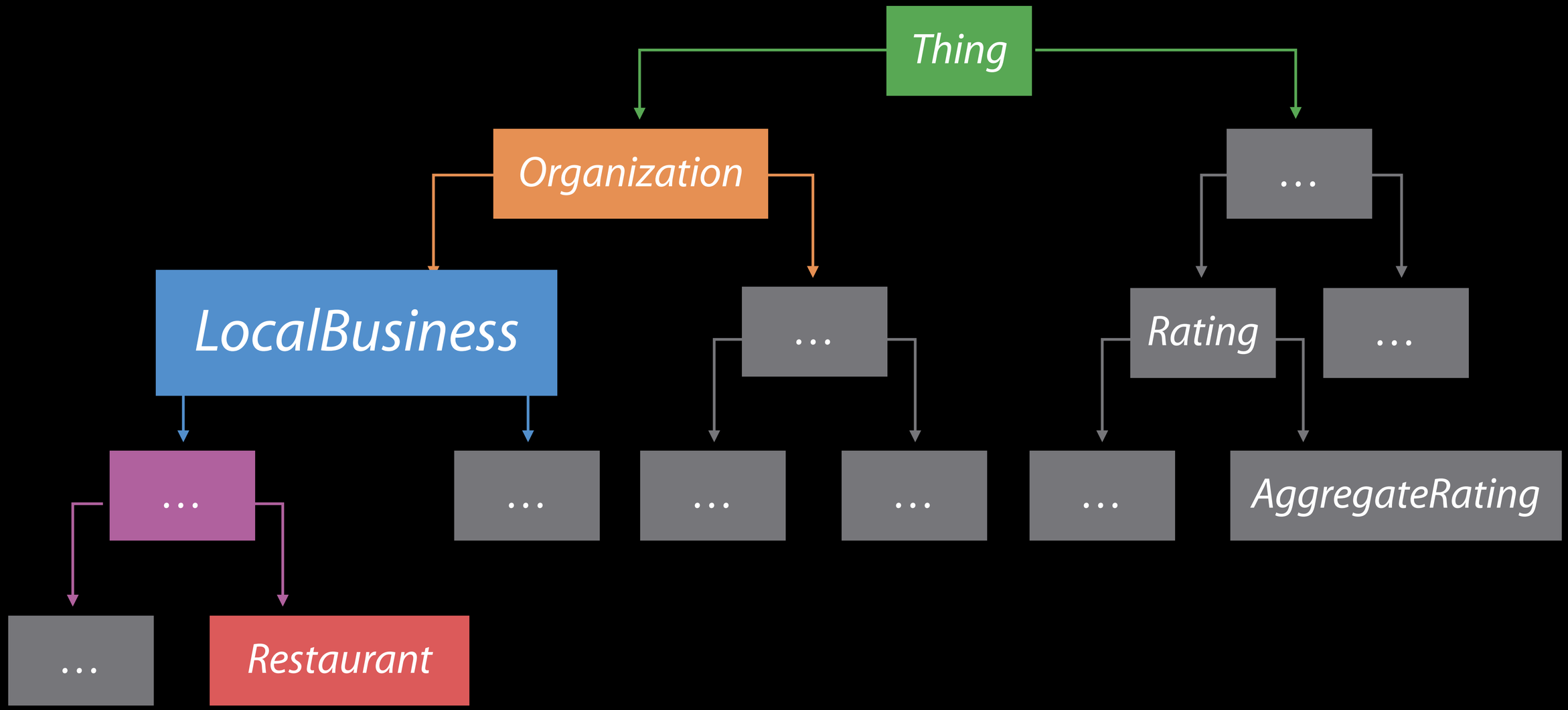
schema.org

Structure



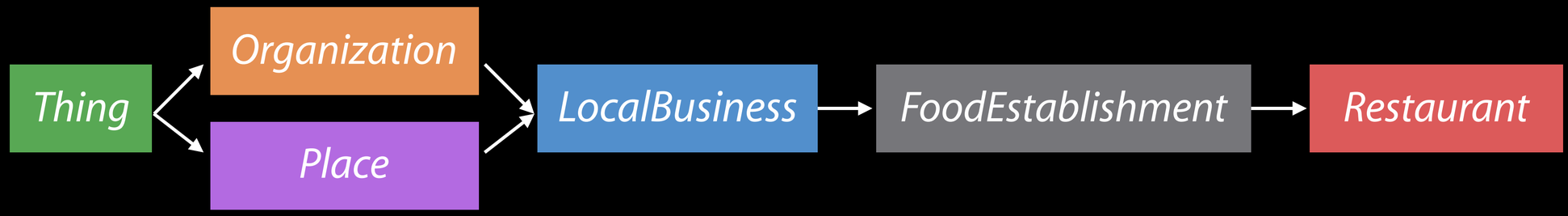
schema.org

Structure



schema.org

Example—Restaurant

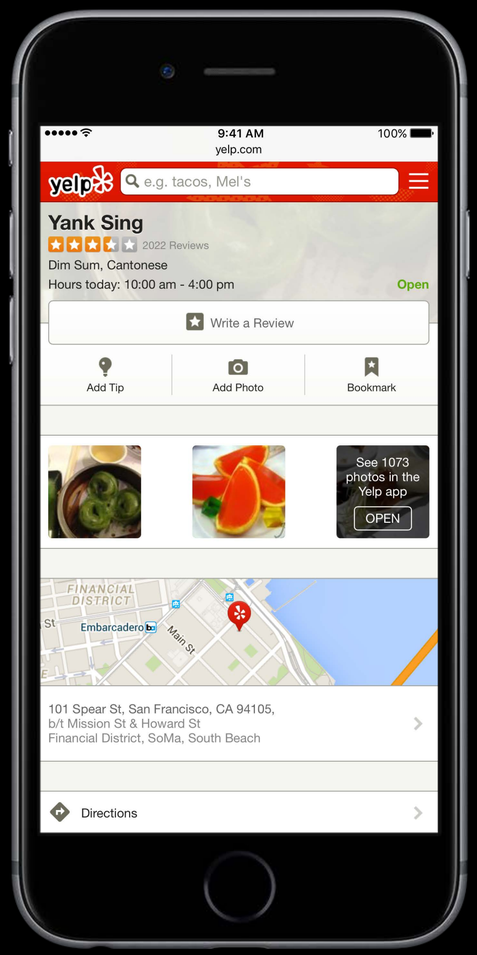


Property	Type	Defined in Schema
name	Text	Thing
aggregateRating	AggregateRating	Organization
address	PostalAddress	LocalBusiness
telephone	Text	LocalBusiness
openingHours	Text	FoodEstablishment
priceRange	Text	FoodEstablishment
acceptsReservations	Boolean	Restaurant
…		

schema.org

Web apps promoting locations

NEW




```
<script type="application/ld+json">
{
    "@context": "http://schema.org",
    "@type": "Restaurant",
    "name": "Yank Sing",
    "telephone": "(415) 781-1111",
    "url": "http://www.yanksing.com",
    "address": {
        "@type": "PostalAddress",
        "streetAddress": "101 Spear St",
        "addressLocality": "San Francisco",
        "postalCode": "94105",
        "addressRegion": "CA"
    },
    "aggregateRating": {
        "@type": "AggregateRating",
        "ratingValue": "3.5",
        "reviewCount": "2022"
    }
}
</script>
```

```
<!-- Restaurant web page without markups -->
<div>
  Yank Sing<br/>
  3.5 stars - based on 2022 reviews<br/>
  101 Spear St, San Francisco, CA 94105<br/>
  (408) 714-1489<br/>
  <a href="http://www.yanksing.com">www.yanksing.com</a>
</div>
```

```
<!-- Restaurant web page without markups -->
<div>
  Yank Sing<br/>
  3.5 stars - based on 2022 reviews<br/>
  101 Spear St, San Francisco, CA 94105<br/>
  (408) 714-1489<br/>
  <a href="http://www.yanksing.com">www.yanksing.com</a>
</div>
```

```
<!-- Restaurant web page with JSON-LD markups -->
<div>
  Yank Sing<br/>
  3.5 stars - based on 2022 reviews<br/>
  101 Spear St, San Francisco, CA 94105<br/>
  (408) 714-1489<br/>
  <a href="http://www.yanksing.com">www.yanksing.com</a>
</div>

<script type="application/ld+json">
{
    "@context": "http://schema.org",
    "@type": "Restaurant",
    "name": "Yank Sing",
    "telephone": "(415) 781-1111",
    "url": "http://www.yanksing.com",
    "address": {
        "@type": "PostalAddress",
        "streetAddress": "101 Spear St",
        "addressLocality": "San Francisco",
        "postalCode": "94105",
        "addressRegion": "CA"
    },
    "aggregateRating": {
        "@type": "AggregateRating",
        "ratingValue": "3.5",
        "reviewCount": "2022"
    }}
</script>
```

```
<!-- Restaurant web page with JSON-LD markups -->
<div>
  Yank Sing<br/>
  3.5 stars - based on 2022 reviews<br/>
  101 Spear St, San Francisco, CA 94105<br/>
  (408) 714-1489<br/>
  <a href="http://www.yanksing.com">www.yanksing.com</a>
</div>

<script type="application/ld+json">
{
    "@context": "http://schema.org",
    "@type": "Restaurant",
    "name": "Yank Sing",
    "telephone": "(415) 781-1111",
    "url": "http://www.yanksing.com",
    "address": {
        "@type": "PostalAddress",
        "streetAddress": "101 Spear St",
        "addressLocality": "San Francisco",
        "postalCode": "94105",
        "addressRegion": "CA"
    },
    "aggregateRating": {
        "@type": "AggregateRating",
        "ratingValue": "3.5",
        "reviewCount": "2022"
    }}
</script>
```

```
<!-- Restaurant web page with JSON-LD markups -->
<div>
  Yank Sing<br/>
  3.5 stars - based on 2022 reviews<br/>
  101 Spear St, San Francisco, CA 94105<br/>
  (408) 714-1489<br/>
  <a href="http://www.yanksing.com">www.yanksing.com</a>
</div>

<script type="application/ld+json">
{
    "@context": "http://schema.org",
    "@type": "Restaurant",
    "name": "Yank Sing",
    "telephone": "(415) 781-1111",
    "url": "http://www.yanksing.com",
    "address": {
        "@type": "PostalAddress",
        "streetAddress": "101 Spear St",
        "addressLocality": "San Francisco",
        "postalCode": "94105",
        "addressRegion": "CA"
    },
    "aggregateRating": {
        "@type": "AggregateRating",
        "ratingValue": "3.5",
        "reviewCount": "2022"
    }}
</script>
```

```
<!-- Restaurant web page with inline microdata markups -->
<div itemscope itemtype="http://schema.org/Restaurant">
  <span itemprop="name">Yank Sing</span>
  <div itemprop="aggregateRating" itemscope itemtype="http://schema.org/AggregateRating">
    <span itemprop="ratingValue">3.5</span> stars -
    based on <span itemprop="reviewCount">2022</span> reviews
  </div>
  <div itemprop="address" itemscope itemtype="http://schema.org/PostalAddress">
    <span itemprop="streetAddress">101 Spear St</span>
    <span itemprop="addressLocality">San Francisco</span>,
    <span itemprop="addressRegion">CA</span> <span itemprop="postalCode">94105</span>
  </div>
  <span itemprop="telephone">(415) 781-1111</span>
  <a itemprop="url" href="http://www.yanksing.com">www.yanksing.com</a>
</div>
```

schema.org

NEW

Web apps promoting locations

Safari relies on location-related schemas for location suggestions

Some of the benefits of NSUserActivity for the web

Support for JSON-LD and Microdata

schema.org

Supported schemas

NEW

Any schemas with:

- PostalAddress
- GeoCoordinates
- A telephone property

Summary

Summary

Promote locations
Contact interactions
Spotlight Search and Handoff

Summary

Promote locations
Contact interactions
Spotlight Search and Handoff

Promote locations on the web

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Summary

1. Location Suggestions in QuickType

2. Routing Apps

1. Location Suggestions in QuickType
2. Routing Apps

NEW

9:41 AM

100%

2135 Jones St

Enter destination

Embarcadero Station
298 Market St, San Francisco, CA

Montgomery St. Station
598 Market St, San Francisco, CA

San Francisco International Airport
San Francisco, CA

Powell St. Station
899 Market St, San Francisco, CA

Union Square
San Francisco, CA

Yank Sing from Yelp
101 Spear St San Francisco CA 94105

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

NEW

9:41 AM 100%

2135 Jones St

Enter destination

Embarcadero Station
298 Market St, San Francisco, CA

Montgomery St. Station
598 Market St, San Francisco, CA

San Francisco International Airport
San Francisco, CA

Powell St. Station
899 Market St, San Francisco, CA

Union Square
San Francisco, CA

Yank Sing from Yelp
101 Spear St San Francisco CA 94105

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

NEW

9:41 AM 100%

2135 Jones St

Enter destination

Embarcadero Station
298 Market St, San Francisco, CA

Montgomery St. Station
598 Market St, San Francisco, CA

San Francisco International Airport
San Francisco, CA

Powell St. Station
899 Market St, San Francisco, CA

Union Square
San Francisco, CA

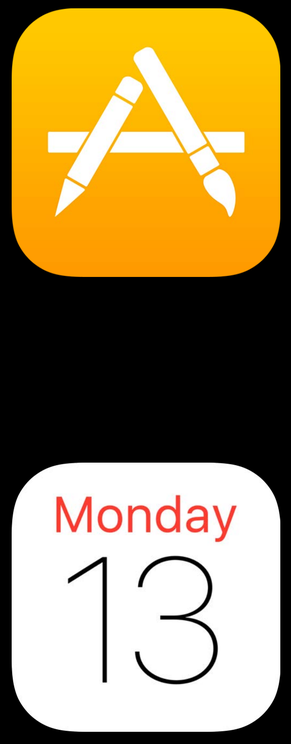
Yank Sing from Yelp
101 Spear St San Francisco CA 94105

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



NEW

9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA
Yank Sing from Yelp
101 Spear St San Francisco CA 94105
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

NEW

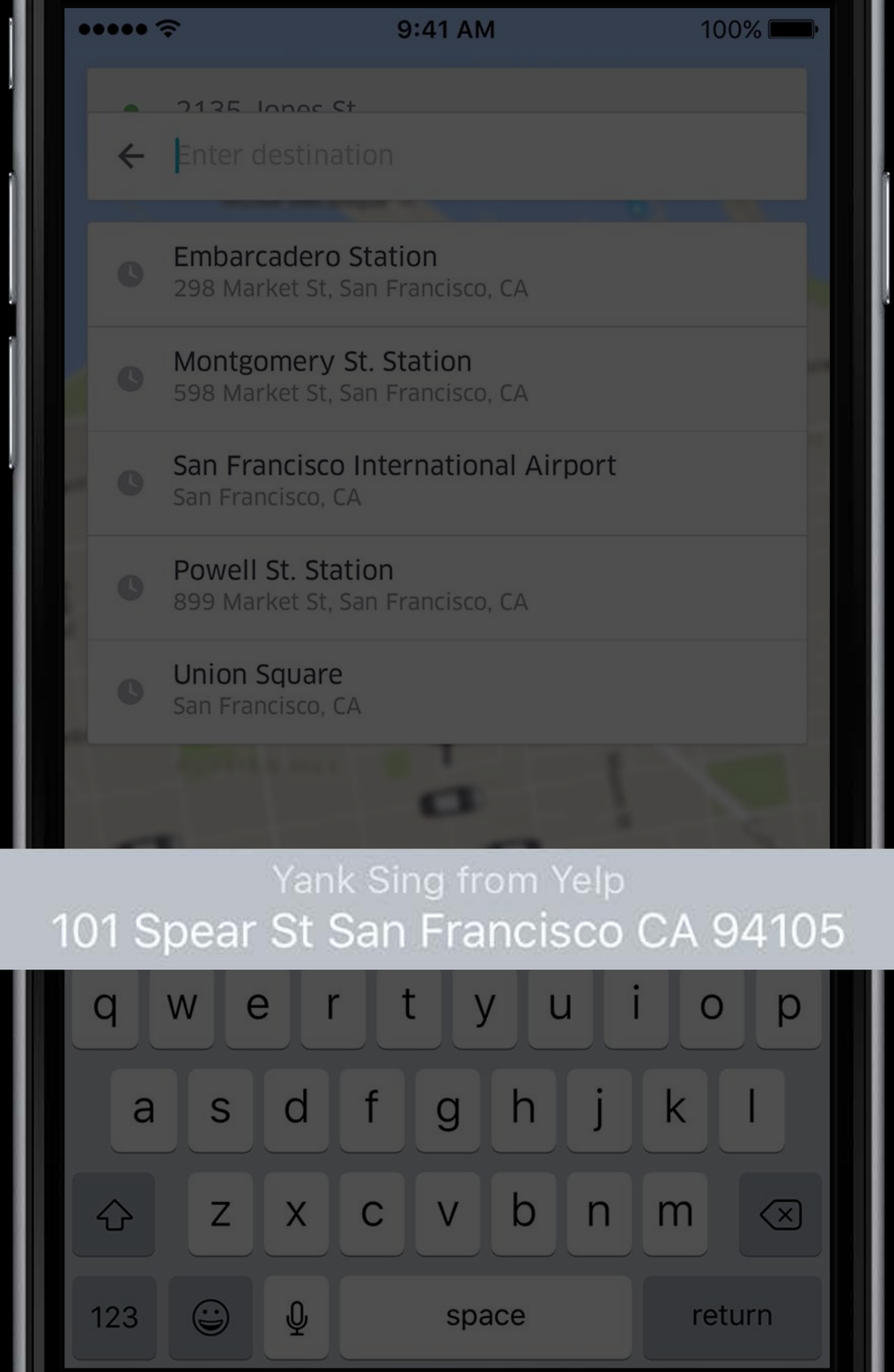
Monday
13

Yank Sing from Yelp
101 Spear St San Francisco CA 94105

9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA


Monday
13

NEW


9:41 AM
100%
Enter destination
Embarcadero Station
298 Market St, San Francisco, CA
Montgomery St. Station
598 Market St, San Francisco, CA
San Francisco International Airport
San Francisco, CA
Powell St. Station
899 Market St, San Francisco, CA
Union Square
San Francisco, CA
Yank Sing from Yelp
101 Spear St San Francisco CA 94105
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

Locations Suggestions

QuickType

NEW

Hint the semantic intent for eligible text containers

Get proactive suggestions in your app

Benefit from enhanced autocorrection

UITextInputTraits

Text content type

NEW

New UITextInputTraits **`textContentType`** API

```
let textField = UITextField()
textField.textContentType = UITextContentTypeFullStreetAddress
```

UITextInputTraits

Text content type

NEW

`UITextContentTypeName`	`UITextContentTypeFullStreetAddress`
`UITextContentTypeGivenName`	`UITextContentTypeAddressCityAndState`
`UITextContentTypeFamilyName`	`UITextContentTypeTelephoneNumber`
`UITextContentTypeLocation`	`UITextContentTypeEmailAddress`

UITextInputTraits

Text content type

NEW

`UITextContentTypeName`	`UITextContentTypeFullStreetAddress`
`UITextContentTypeGivenName`	`UITextContentTypeAddressCityAndState`
`UITextContentTypeFamilyName`	`UITextContentTypeTelephoneNumber`
`UITextContentTypeLocation`	`UITextContentTypeEmailAddress`

UITextInputTraits

Text content type

NEW

`UITextContentTypeName`	`UITextContentTypeFullStreetAddress`
`UITextContentTypeGivenName`	`UITextContentTypeAddressCityAndState`
`UITextContentTypeFamilyName`	`UITextContentTypeTelephoneNumber`
`UITextContentTypeLocation`	`UITextContentTypeEmailAddress`

UITextInputTraits

Text content type

NEW

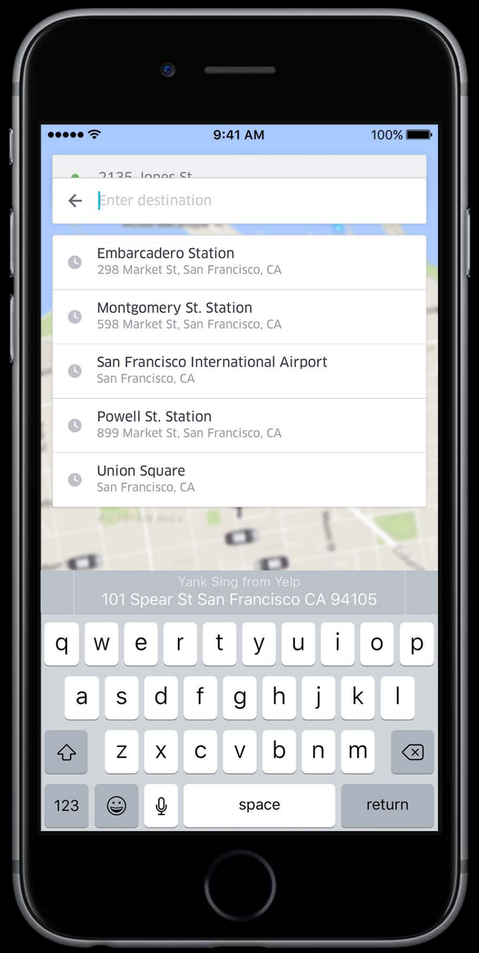
`UITextContentTypeName`	`UITextContentTypeFullStreetAddress`
`UITextContentTypeGivenName`	`UITextContentTypeAddressCityAndState`
`UITextContentTypeFamilyName`	`UITextContentTypeTelephoneNumber`
`UITextContentTypeLocation`	`UITextContentTypeEmailAddress`

```
UITextContentTypeFullStreetAddress
UITextContentTypeCityAndState
```

Location Suggestions

Granularity

NEW




```
UITextContentTypeFullStreetAddress
```

Location Suggestions

Granularity

NEW


```
UITextContentTypeFullStreetAddress
```

Location Suggestions

Granularity

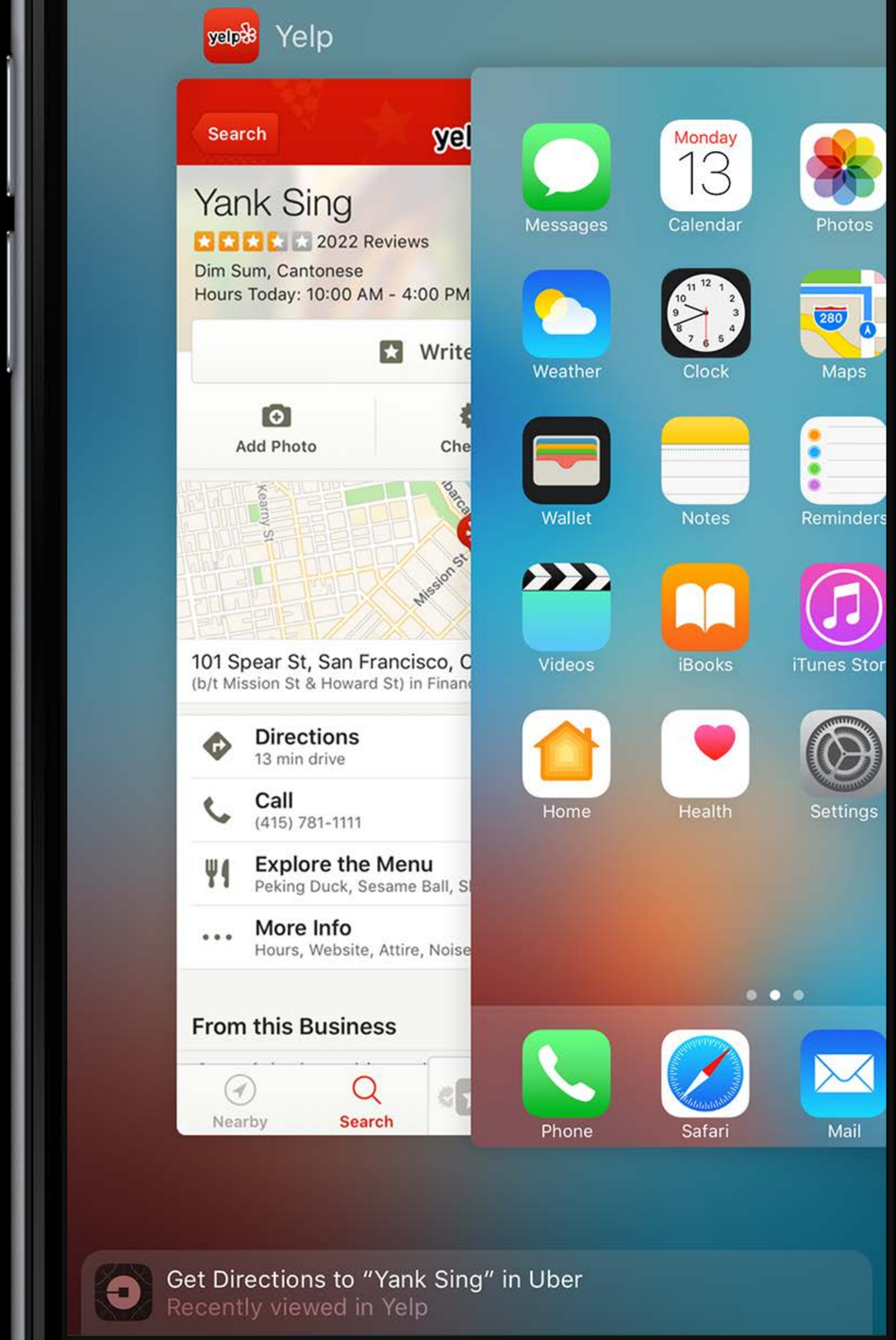
NEW

`UITextContentType`**`FullStreetAddress`**

`UITextContentType`**`AddressCityAndState`**

1. Location Suggestions in QuickType
2. Routing Apps

- Location Suggestions in QuickType
2. Routing Apps


Yelp
Search
Yank Sing
2022 Reviews
Dim Sum, Cantonese
Hours Today: 10:00 AM - 4:00 PM
Write
Add Photo
101 Spear St, San Francisco, C
(b/t Mission St & Howard St) in Financ
Directions
13 min drive
Call
(415) 781-1111
Explore the Menu
Peking Duck, Sesame Ball, S
More Info
Hours, Website, Attire, Noise
From this Business
Nearby
Search
Messages
Monday
13
Calendar
Photos
Weather
Clock
Maps
Wallet
Notes
Reminders
Videos
iBooks
iTunes Stor
Home
Health
Settings
Phone
Safari
Mail
Get Directions to "Yank Sing" in Uber
Recently viewed in Yelp

Explore the Menu
Peking Duck, Sesame Ball, S

More Info
Hours, Website, Attire, Noise

From this Business

Nearby

Search

Phone

Safari

Mail

Get Directions to “Yank Sing” in Uber
Recently viewed in Yelp

Consuming Locations

For routing apps

NEW

Promote your app in Multitasking

Register as a routing app

Handle launch with MapKit's MKDirectionsRequest

iOS learns to suggest your app based on user engagement

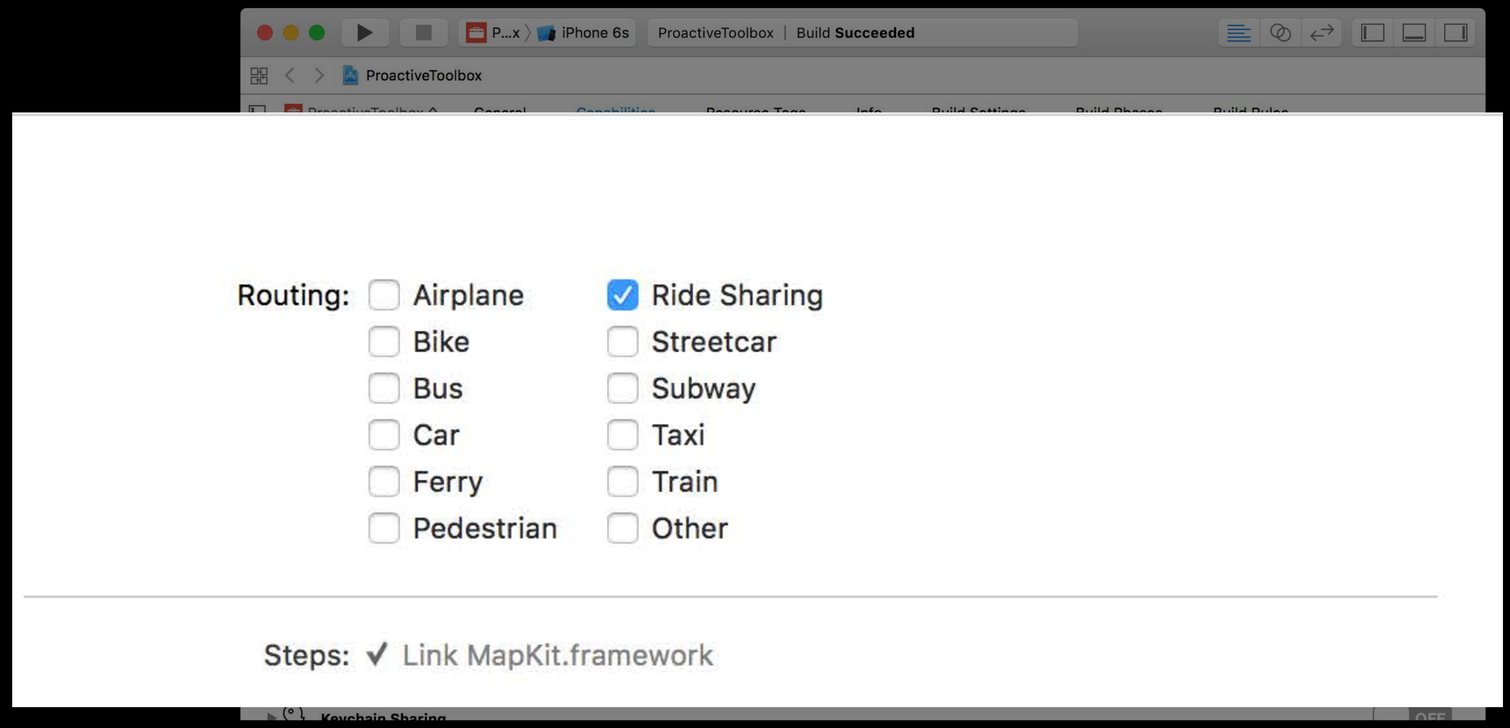
MKDirectionsRequest

Overview

MKDirectionsRequest

Overview

Configure your app to accept directions requests in Xcode

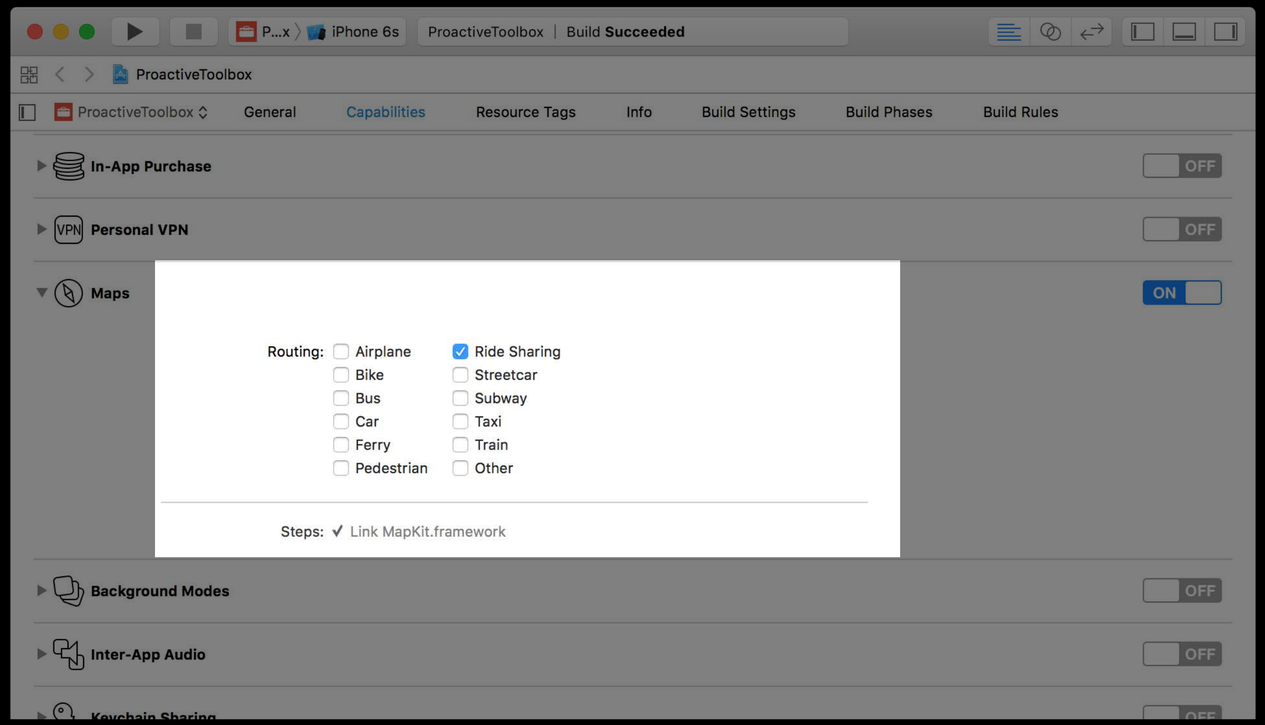


MKDirectionsRequest

Overview

Configure your app to accept directions requests in Xcode

Declare the map regions that your app supports



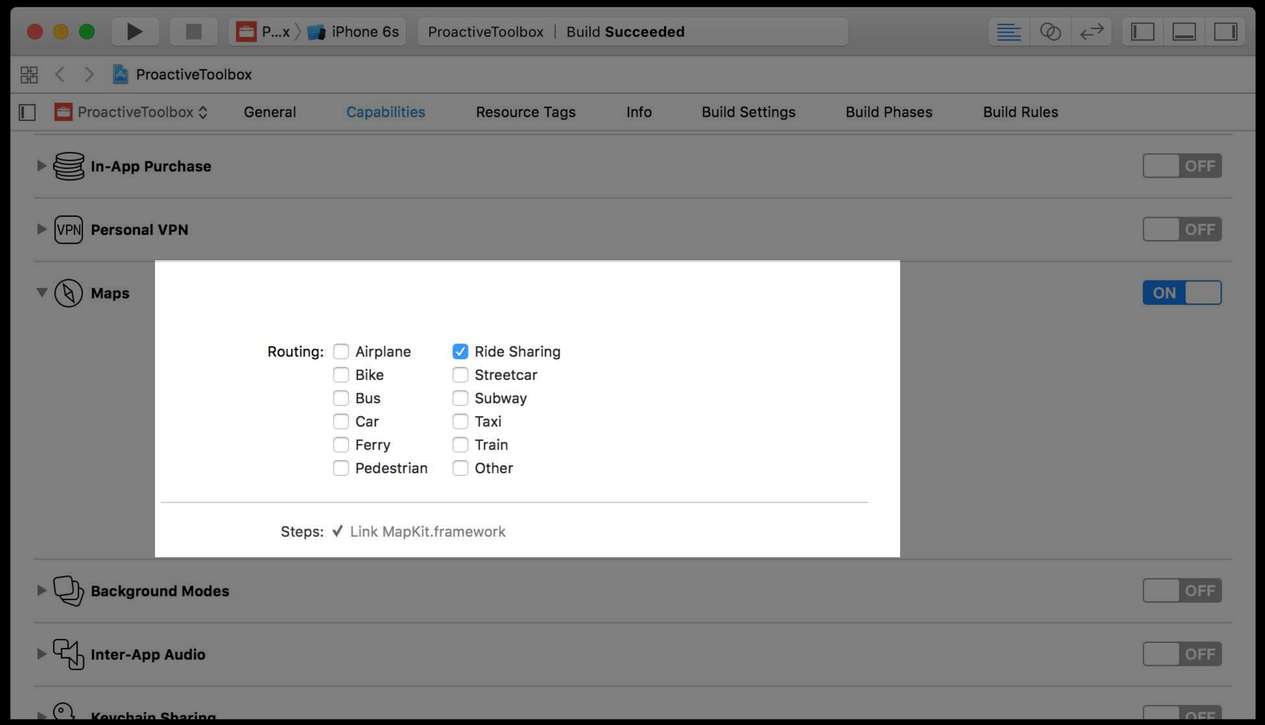
MKDirectionsRequest

Overview

Configure your app to accept directions requests in Xcode

Declare the map regions that your app supports

Process direction request URLs when they are sent to your app



NEW

```swift
// Handle launch with MKDirectionsRequest
func application(app: UIApplication,
                 url: NSURL,
                 options: [String : AnyObject]) -> Bool {
  if (MKDirectionsRequest.isDirectionsRequestURL(url)) {
     let directionsRequest = MKDirectionsRequest(contentsOfURL: url)
     // Handle routing request
     guard
       let coordinate = directionsRequest.destination?.placemark.location?.coordinate,
       let addressDictionary = directionsRequest.destination?.placemark.addressDictionary
       where !CLLocationCoordinate2DIsValid(coordinate)
     else { return true }
     let geocoder = CLGeocoder()
     geocoder.geocodeAddressDictionary(addressDictionary, completionHandler: { (place, err) in
       // Handle launch
     })
   }
   return true
}
```

NEW

```
// Handle launch with MKDirectionsRequest
func application(app: UIApplication,
                 url: NSURL,
                 options: [String : AnyObject]) -> Bool {
  if (MKDirectionsRequest.isDirectionsRequestURL(url)) {
    let directionsRequest = MKDirectionsRequest(contentsOfURL: url)
    // Handle routing request
    guard
      let coordinate = directionsRequest.destination?.placemark.location?.coordinate,
      let addressDictionary = directionsRequest.destination?.placemark.addressDictionary
      where !CLLocationCoordinate2DIsValid(coordinate)
    else { return true }
    let geocoder = CLGeocoder()
    geocoder.geocodeAddressDictionary(addressDictionary, completionHandler: { (place, err) in
      // Handle launch
    })
  }
  return true
}
```

NEW

```
// Handle launch with MKDirectionsRequest
func application(app: UIApplication,
                 url: NSURL,
                 options: [String : AnyObject]) -> Bool {
  if (MKDirectionsRequest.isDirectionsRequestURL(url)) {
    let directionsRequest = MKDirectionsRequest(contentsOfURL: url)
    // Handle routing request
    guard
      let coordinate = directionsRequest.destination?.placemark.location?.coordinate,
      let addressDictionary = directionsRequest.destination?.placemark.addressDictionary
      where !CLLocationCoordinate2DIsValid(coordinate)
    else { return true }
    let geocoder = CLGeocoder()
    geocoder.geocodeAddressDictionary(addressDictionary, completionHandler: { (place, err) in
      // Handle launch
    })
  }
  return true
}
```

NEW

```swift
// Handle launch with MKDirectionsRequest
func application(app: UIApplication,
                 url: NSURL,
                 options: [String : AnyObject]) -> Bool {
  if (MKDirectionsRequest.isDirectionsRequestURL(url)) {
     let directionsRequest = MKDirectionsRequest(contentsOfURL: url)
     // Handle routing request
     guard
       let coordinate = directionsRequest.destination?.placemark.location?.coordinate,
       let addressDictionary = directionsRequest.destination?.placemark.addressDictionary
       where !CLLocationCoordinate2DIsValid(coordinate)
     else { return true }
     let geocoder = CLGeocoder()
     geocoder.geocodeAddressDictionary(addressDictionary, completionHandler: { (place, err) in
       // Handle launch
     })
  }
  return true
}
```

- Location Suggestions in QuickType
- 2 Routing Apps

- Location Suggestions in QuickType
- Routing Apps

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Summary

Promoting Your Media App

Overview

Promoting Your Media App

Overview

iOS promotes the app a user is likely to use based on behavior

Promoting Your Media App

Overview

iOS promotes the app a user is likely to use based on behavior

Suggestions are offered in Spotlight and Today View

Promoting Your Media App

Overview

iOS promotes the app a user is likely to use based on behavior

Suggestions are offered in Spotlight and Today View

If they follow a particular trigger, suggestions may be elevated to the lock screen

Promoting Your Media App

Overview

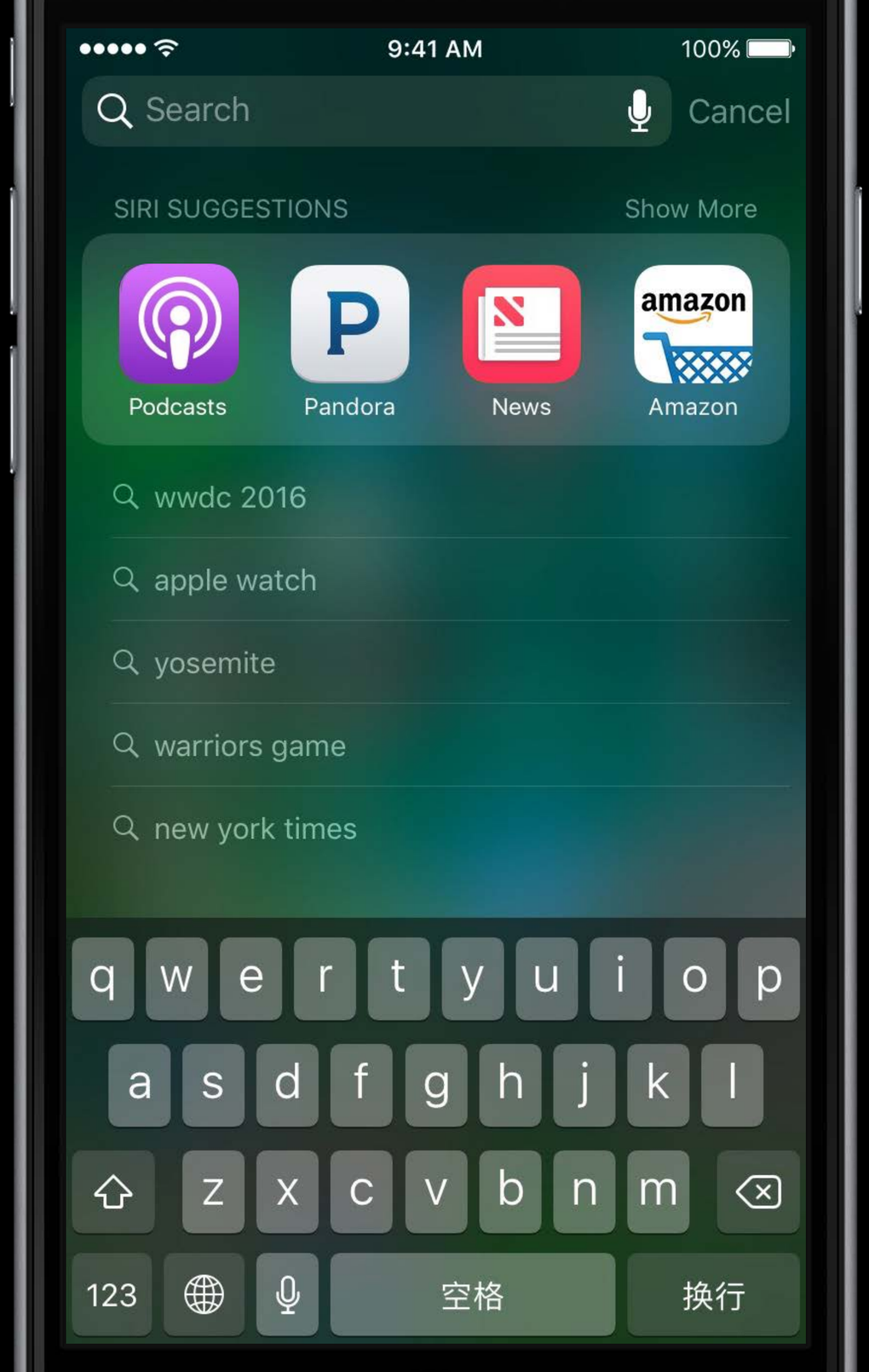
iOS promotes the app a user is likely to use based on behavior

Suggestions are offered in Spotlight and Today View

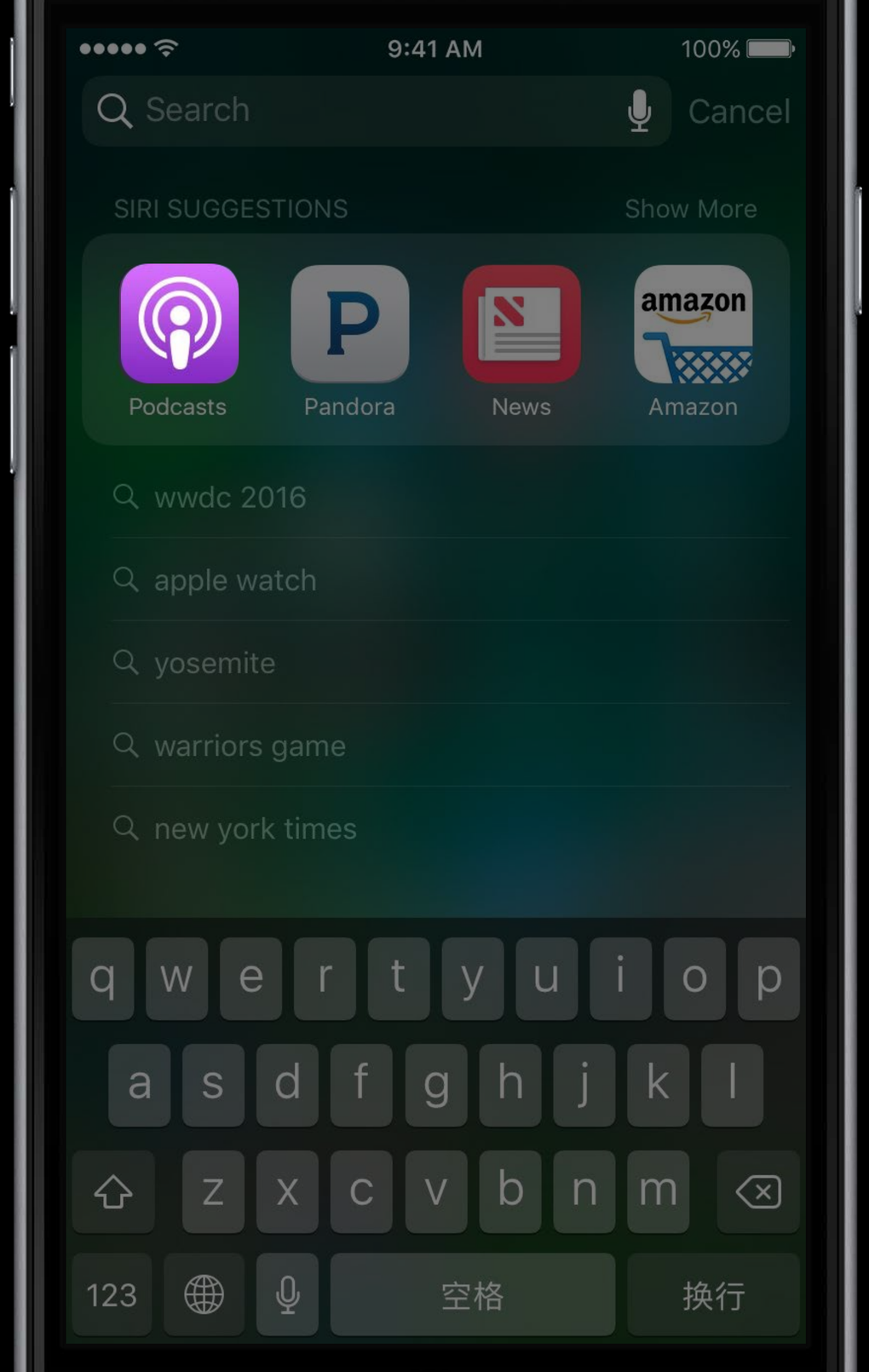
If they follow a particular trigger, suggestions may be elevated to the lock screen

For example:

- After plugging in headphones or a BT device (Podcasts, Music)
- After getting in a car (Maps)
- After arriving at home or work (Agenda app)


9:41 AM
100%
Search
Cancel
SIRI SUGGESTIONS
Show More
Podcasts
Pandora
News
Amazon
wwdc 2016
apple watch
yosemite
warriors game
new york times
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
空格
换行

Spotlight


9:41 AM
100%
Search
Cancel
SIRI SUGGESTIONS
Show More
Podcasts
Pandora
News
Amazon
wwdc 2016
apple watch
yosemite
warriors game
new york times
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
空格
换行

Spotlight

Lockscreen

With a simple API, you can build a far more engaging lock screen experience.

Before


9:41 AM
100%
3:29
-56:31
The Power Of Design
TED Radio Hour — May 19, 2016 at
15
15
npr
TED
RADIO
HOUR
Press home to open

After

Before


9:41 AM
100%
3:29
-56:31
The Power Of Design
TED Radio Hour — May 19, 2016 at
15
15
npr
TED
RADIO
HOUR
Press home to open

After

10x

Increase in conversion

How?

MPPlayableContentManager

```
struct MediaItem {
    var title: String
    var artist: String
    var album: String
    var duration: Double
    var albumArtwork: UIImage
    var albumArtworkSize: CGSize
}
```

```swift
import MediaPlayer

class MyPlayer: MPPlayableContentDelegate, NSObject {
    override init() {
        super.init()
        MPPlayableContentManager.shared().delegate = self
    }
}
```

```
import MediaPlayer

class MyPlayer: MPPlayableContentDelegate, NSObject {
    override init() {
        super.init()
        MPPlayableContentManager.shared().delegate = self
    }
}
```

```swift
import MediaPlayer

class MyPlayer: MPPlayableContentDelegate, NSObject {
    override init() {
        super.init()
        MPPlayableContentManager.shared().delegate = self
    }
}
```

```swift
import MediaPlayer

class MyPlayer: MPPlayableContentDelegate, NSObject {
    override init() {
        super.init()
        MPPlayableContentManager.shared().delegate = self
    }
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```
    func playableContentManager(contentManager: MPPlayableContentManager,
                                initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                                completionHandler: (NSError?) -> Void) {

        guard let mediaItem = findMediaItemToPlay() else {
            // Handle error, call completion handler with error and return
        }
        populateNowPlayingInfo(mediaItem: mediaItem)

        guard let status = preparePlayback(mediaItem: mediaItem) else {
            // Handle error, call completion handler with error and return
        }

        // Do other things that are required for success

        completionHandler(nil)
    }
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
           MPMediaItemPropertyTitle: item.title,
           MPMediaItemPropertyArtist: item.artist,
           MPMediaItemPropertyAlbumTitle: item.album,
           MPMediaItemPropertyPlaybackDuration: item.duration,
           MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```
func populateNowPlayingInfo(item: MediaItem) {
    let infoCenter = MPNowPlayingInfoCenter.defaultCenter()
    if let image = item.albumArtwork ?? UIImage(named: "EmptyAlbum") {
       let albumArt = MPMediaItemArtwork(boundsSize: item.albumArtworkSize,
                                         requestHandler: { (size) -> UIImage in
            return image
       })

       let nowPlayingInfo = [
            MPMediaItemPropertyTitle: item.title,
            MPMediaItemPropertyArtist: item.artist,
            MPMediaItemPropertyAlbumTitle: item.album,
            MPMediaItemPropertyPlaybackDuration: item.duration,
            MPMediaItemPropertyArtwork: albumArt
       ]

       infoCenter.nowPlayingInfo = nowPlayingInfo
    }
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

```swift
func playableContentManager(contentManager: MPPlayableContentManager,
                            initializePlaybackQueueWithContentItems contentItems: [AnyObject]?,
                            completionHandler: (NSError?) -> Void) {

    guard let mediaItem = findMediaItemToPlay() else {
        // Handle error, call completion handler with error and return
    }
    populateNowPlayingInfo(mediaItem: mediaItem)

    guard let status = preparePlayback(mediaItem: mediaItem) else {
        // Handle error, call completion handler with error and return
    }

    // Do other things that are required for success

    completionHandler(nil)
}
```

Promoting Your Media App

Overview

iOS promotes the app a user is likely to use based on behavior

Suggestions are offered in Spotlight and Today View

If they follow a particular trigger, suggestions may be elevated to the lock screen

For example:

- After plugging in headphones or a BT device (Podcasts, Music)
- After getting in a car (Maps)
- After arriving at home or work (Agenda app)

With a simple API, you can build a far more engaging lock screen experience

NSUserActivity and schema.org

Location Suggestions

Media App Suggestions

Summary

Proactive Suggestions

Summary

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

- NSUserActivity—The "eyes" of the OS

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

- NSUserActivity—The "eyes" of the OS
- schema.org—Like NSUserActivity, but for the web

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

- NSUserActivity—The "eyes" of the OS
- schema.org—Like NSUserActivity, but for the web
- MKDirectionsRequest and a new UIKit API—Build a seamless experience for handling locations

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

- NSUserActivity—The "eyes" of the OS
- schema.org—Like NSUserActivity, but for the web
- MKDirectionsRequest and a new UIKit API—Build a seamless experience for handling locations
- MPPlayableContentManager—Enhanced experience for media suggestions

Proactive Suggestions

Summary

Simple APIs that deeply integrate your app to the OS

- NSUserActivity—The "eyes" of the OS
- schema.org—Like NSUserActivity, but for the web
- MKDirectionsRequest and a new UIKit API—Build a seamless experience for handling locations
- MPPlayableContentManager—Enhanced experience for media suggestions

Easy to adopt, easy to test

More Information

https://developer.apple.com/wwdc16/240

Related Sessions

Introducing SiriKit	Presidio	Wednesday 5:00PM
Making the Most of Search APIs	Pacific Heights	Thursday 11:00AM
Extending Your Apps with SiriKit	Presidio	Thursday 1:40PM
Adopting Handoff on iOS and OS X		WWDC 2014
Introducing Search APIs		WWDC 2015

Labs

Proactive Suggestions Lab	Frameworks Lab A	Friday 3:00PM

The more the system knows about your app,

The more the system knows about your app, the more opportunity it will have to promote it.

WWDC16