Introduction to Siri Shortcuts

Session 211

Ari Weinstein, Siri
Willem Mattelaer, Siri
Shortcuts let you expose the capabilities of your apps to Siri.
Order Mint Mojito
Order a large, medium cream, medium sugar Mint Mojito at Philz Cupertino.
Order Mint Mojito
Order a large, medium cream, medium sugar Mint Mojito at Philz Cupertino.
Ready to order with Philz Coffee?

Pick-up at Cupertino - 10:00am

Is it Mint Mojito Time?
Large, Medium Cream, Medium Sugar

No  Yes
Thanks Kim!
See you at Cupertino in 11 mins

Mint Mojito
Large, Medium Cream, Medium Sugar

$5.50

OK. Ordering. Philz Coffee says: "Great! Kim, we'll see you at Philz Cupertino in 11 minutes for pick-up."
Add to Siri

Order Mint Mojito
Order a large, medium cream, medium sugar Mint Mojito at Philz Cupertino.

Philz Coffee

You can say something like...

“Coffee time”

After you record your personalized phrase, Siri can use it to tell “Philz Coffee” run this shortcut.
Adopting shortcuts
Optimizing for suggestions
Privacy
Media shortcuts
Adopting Shortcuts
Creating Shortcuts
Creating Shortcuts

Define Shortcut
Creating Shortcuts

Define Shortcut → Donate Shortcut
Creating Shortcuts

Define Shortcut → Donate Shortcut → Handle Shortcut
What Makes a Great Shortcut?

Every shortcut should:
- Accelerate the user to perform a key function of your app
- Be of persistent interest to the user
- Be executable at any time
Shortcuts APIs

NSUserActivity

Intents
Choosing an Adoption Strategy

Use NSUserActivity if your shortcut:
• Opens something in your app
• Represents showing items that you index in Spotlight or offer for Handoff
Choosing an Adoption Strategy

For the best experience, adopt Intents. Intents-based shortcuts can:

- Run inline, without launching your app
- Include custom voice response or a custom UI
- Include granular predictions
NSUserActivity
NSUserActivity
Define your shortcut

 Declare a type in NSUserActivityTypes in your Info.plist

```xml
<key>NSUserActivityTypes</key>
<array>
  <string>com.myapp.name.my-activity-type</string>
</array>
```
There is a code snippet related to creating a user activity in a specific activity type. The code initializes a `userActivity` with the activity type, sets its eligibility for search and prediction, and includes a title and `userInfo` dictionary. It also sets a suggested invocation phrase. Additionally, it creates a `CSSearchableItemAttributeSet` object with a thumbnail image and content description, which is then associated with the `userActivity`. Finally, the `userActivity` is assigned to a view controller.
let userActivity = NSUserActivity(activityType: "com.myapp.name.my-activity-type")
userActivity.isEligibleForSearch = true
userActivity.isEligibleForPrediction = true
userActivity.title = "Activity"
userActivity.userInfo = ["key": "value"]
userActivity.suggestedInvocationPhrase = "Let's do it"

let attributes = CSSearchableItemAttributeSet(itemContentType: kUTTypeItem as String)
let image = UIImage(named: "myImage")!
attributes.thumbnailData = image.pngData()
attributes.contentDescription = "Subtitle"
userActivity.contentAttributeSet = attributes

viewController.userActivity = userActivity
// How to donate a shortcut

```swift
let userActivity = NSUserActivity(activityType: "com.myapp.name.my-activity-type")
userActivity.isEligibleForSearch = true
userActivity.isEligibleForPrediction = true
userActivity.title = "Activity"
userActivity.userInfo = ["key": "value"]
userActivity.suggestedInvocationPhrase = "Let's do it"

let attributes = CSSearchableItemAttributeSet(itemContentType: kUTTypeItem as String)
let image = UIImage(named: "myImage")!
attributes.thumbnailData = image.pngData()
attributes.contentDescription = "Subtitle"
userActivity.contentAttributeSet = attributes

viewController.userActivity = userActivity
```
// How to donate a shortcut

let userActivity = NSUserActivity(activityType: "com.myapp.name.my-activity-type")
userActivity.isEligibleForSearch = true
userActivity.isEligibleForPrediction = true
userActivity.title = "Activity"
userActivity.userInfo = ["key": "value"]
userActivity.suggestedInvocationPhrase = "Let's do it"

let attributes = CSSearchableItemAttributeSet(itemContentType: kUTTypeItem as String)
let image = UIImage(named: "myImage")!
attributes.thumbnailData = image.pngData() 
attributes.contentDescription = "Subtitle"
userActivity.contentAttributeSet = attributes

viewController.userActivity = userActivity
// How to donate a shortcut

let userActivity = NSUserActivity(activityType: "com.myapp.name.my-activity-type")
userActivity.isEligibleForSearch = true
userActivity.isEligibleForPrediction = true
userActivity.title = "Activity"
userActivity.userInfo = ["key": "value"]
userActivity.suggestedInvocationPhrase = "Let's do it"

let attributes = CSSearchableItemAttributeSet(itemContentType: kUTTypeItem as String)
let image = UIImage(named: "myImage")!
attributes.thumbnailData = image.pngData()
attributes.contentDescription = "Subtitle"
userActivity.contentAttributeSet = attributes

viewController.userActivity = userActivity
func application(_ application: UIApplication, continue userActivity: NSUserActivity, restorationHandler: @escaping ([Any]?) -> Void) -> Bool {
    if userActivity.activityType == "com.myapp.name.my-activity-type" {
        // Restore state for userActivity and userInfo
    }
}
Intents
Custom

Built-In

- Messaging
- Climate and Radio
- Workouts
- Photo Search
- Ride Booking
- Notebook
- Lists
- Payments
- Car Commands
- VOIP Calling
Intents
Define your shortcut

Create an intent definition file in Xcode to customize a built-in SiriKit intent or define a new custom intent.

Go to File -> New File, and choose “SiriKit Intent Definition File.”
### Custom Intent

**Category:** Do

**Title:**

**Description:**

- **Confirmation:** User confirmation required

### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Array</th>
</tr>
</thead>
</table>

**+**

**-**

### Shortcut Types

**Parameter Combination:** (No Parameters)

<table>
<thead>
<tr>
<th>Title</th>
<th>Subtitle</th>
</tr>
</thead>
</table>

- **Background:** Supports background execution
Custom Intent

- **Category**: Order
- **Title**: Order Soup
- **Description**: Order a soup to a specific location
- **Confirmation**: User confirmation required

**Parameters**
- Items
- deliveryLocation

**Shortcut Types**

<table>
<thead>
<tr>
<th>Parameter Combination</th>
<th>Title</th>
<th>Subtitle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items, deliveryLocation</td>
<td>Items</td>
<td>deliveryLocation</td>
</tr>
</tbody>
</table>

**Background**: Supports background execution
Custom Intent

<table>
<thead>
<tr>
<th>Category</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Order Soup</td>
</tr>
<tr>
<td>Description</td>
<td>Order a soup to a specific location</td>
</tr>
<tr>
<td>Confirmation</td>
<td>User confirmation required</td>
</tr>
</tbody>
</table>

Shortcut Types

<table>
<thead>
<tr>
<th>Parameter Combination</th>
<th>Title</th>
<th>Subtitle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items, deliveryLocation</td>
<td>Order Items</td>
<td>To deliveryLocation</td>
</tr>
</tbody>
</table>

Background: Supports background execution
**Custom Intent**

**Category:** Order

**Title:** Order Soup

**Description:** Order a soup to a specific location

**Confirmation:** User confirmation required

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>Custom</td>
<td></td>
</tr>
<tr>
<td>deliveryLocation</td>
<td>Location</td>
<td></td>
</tr>
</tbody>
</table>

**Shortcut Types**

**Parameter Combination**
- Items
- Items, deliveryLocation
- deliveryLocation

**Title:** Order Items

**Subtitle:** To deliveryLocation

**Background:** Supports background execution
CUSTOM INTENTS
- OrderSoup
- Response

Custom Intent
- Category: Order
- Title: Order Soup
- Description: Order a soup to a specific location
- Confirmation: User confirmation required

Parameters
- Items: Custom
- deliveryLocation: Location

Subtitle: To deliveryLocation

Background: Supports background execution
**Custom Intents**

**OrderSoup**

**Custom Intent**

- **Category**: Order
- **Title**: Order Soup
- **Description**: Order a soup to a specific location
- **Confirmation**: User confirmation required

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>Custom</td>
<td>✔</td>
</tr>
<tr>
<td>deliveryLocation</td>
<td>Location</td>
<td>✔</td>
</tr>
</tbody>
</table>

**Shortcut Types**

- **Parameter Combination**: Items, deliveryLocation
  - **Title**: Order Items
  - **Subtitle**: To deliveryLocation
  - **Background**: Supports background execution
Custom Intent

Category: Order
Title: Order Soup
Description: Order a soup to a specific location
Confirmation: User confirmation required

Shortcut Types

Parameter Combination: items, deliveryLocation
Title: Order items
Subtitle: To deliveryLocation
Background: Supports background execution
public class OrderSoupIntent: INIntent {
    public var items: [INObject]?
    public var deliveryLocation: CLPlacemark?
}

public protocol OrderSoupIntentHandling: NSObjectProtocol {
    public func handle(intent: OrderSoupIntent, completion: @escaping 
    (OrderSoupIntentResponse) -> Void)
    optional func confirm(intent: OrderSoupIntent, completion: @escaping 
    (OrderSoupIntentResponse) -> Void)
}

Intents
Define your shortcut

Target selection

• Your intent definition should be added to any target where the intents are used
• If you're using a framework, only generate classes for the framework
// Donate your shortcut

let intent = PlaceOrderIntent()
intent.items = order.items.map({ INObject(identifier: $0.id.uuidString, display: $0.menuItem.displayName) })
intent.deliveryLocation = order.destinationLocation

let interaction = INInteraction(intent: intent, response: nil)
interaction.donate { error in
    // Handle error
}
// Handle your shortcut

func application(_ application: UIApplication, continue userActivity: NSUserActivity, restorationHandler: @escaping ([Any]?) -> Void) -> Bool {
    if userActivity.activityType == "OrderSoupIntent",
        let intent = userActivity.interaction?.intent as? OrderSoupIntent {
            // Show ordering UI, pre-populated with the fields from the intent
        }
    }
}
// Handle your shortcut

class IntentHandler: INExtension, OrderSoupIntentHandling {

    func confirm(intent: OrderSoupIntent, completion: (OrderSoupIntentResponse) -> Void) {
        completion(OrderSoupIntentResponse(code: .ready, userActivity: nil))
    }

    func handle(intent: OrderSoupIntent, completion: (OrderSoupIntentResponse) -> Void) {
        // Order the soup
        completion(OrderSoupIntentResponse(code: .success, userActivity: nil))
    }
}
Intent Handling Notes

Implement an intents extension to run your shortcut in the background, without launching your app.

Always implement \texttt{-continueUserActivity}, even if you have an intents extension.
Expose Shortcuts to the Siri Watch Face by providing INRelevantShortcut objects.

Optionally, include relevance information as a hint as to when your shortcuts should be shown on the Watch Face.

Works even if you don’t have a Watch app.
Demo

Adopting Shortcuts

Willem Mattelaer, Siri
Soup Chef

Made NSUserActivity eligible for prediction
Soup Chef

Made NSUserActivity eligible for prediction

Defined a custom intent
Soup Chef

Made NSUserActivity eligible for prediction

Defined a custom intent

Donated the intent
Soup Chef

Made NSUserActivity eligible for prediction
Defined a custom intent
Donated the intent
Handled the intent
Optimizing for Suggestions
How is a Shortcut Suggested?
How is a Shortcut Suggested?

Order Tomato Soup
How is a Shortcut Suggested?

Order Tomato Soup
How is a Shortcut Suggested?
How is a Shortcut Suggested?

Order tomato soup with croutons

Monday, 11:57 AM
How is a Shortcut Suggested?

Order tomato soup with croutons

Order tomato soup with red pepper

Time

Monday, 11:57 AM

Monday, 6:06 PM
How is a Shortcut Suggested?

- Monday, 11:57 AM: Order tomato soup with croutons
- Monday, 6:06 PM: Order tomato soup with red pepper
- Tuesday, 11:59 AM: Order tomato soup with croutons
How is a Shortcut Suggested?

Order tomato soup with croutons
Order tomato soup with red pepper
Order tomato soup with croutons...

Time

Monday, 11:57 AM
Monday, 6:06 PM
Tuesday, 11:59 AM

Shortcut Suggestion

Friday, 12:00 PM
How is a Shortcut Suggested?

Order tomato soup with croutons
Order tomato soup with red pepper
Order tomato soup with croutons
...
Order tomato soup with croutons
How is a Shortcut Suggested?

NSUserActivity

NSUserActivity.userInfo

- soup
- quantity
- scrollPosition
How is a Shortcut Suggested?

NSUserActivity
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>NSUserActivity</th>
<th>&quot;tomato&quot;</th>
<th>1</th>
<th>79.0</th>
</tr>
</thead>
</table>

Time
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>NSUserActivity</th>
<th>&quot;tomato&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>79.0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NSUserActivity</th>
<th>&quot;tomato&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>110.0</td>
<td></td>
</tr>
</tbody>
</table>
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>Time</th>
<th>Activity</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>79.0</td>
<td>tomato</td>
<td>1</td>
</tr>
<tr>
<td>110.0</td>
<td>tomato</td>
<td>1</td>
</tr>
<tr>
<td>45.0</td>
<td>tomato</td>
<td>1</td>
</tr>
</tbody>
</table>
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>Activity</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>tomato</td>
<td>79.0</td>
</tr>
<tr>
<td></td>
<td>110.0</td>
</tr>
<tr>
<td></td>
<td>45.0</td>
</tr>
</tbody>
</table>

...
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>Time</th>
<th>NSUserActivity</th>
<th>Activity Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“tomato”</td>
<td></td>
</tr>
<tr>
<td>79.0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>“tomato”</td>
<td></td>
</tr>
<tr>
<td>110.0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>“tomato”</td>
<td></td>
</tr>
<tr>
<td>45.0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Shortcut Suggestion
How is a Shortcut Suggested?

NSUserActivity

requiredUserInfoKeys

• Minimal amount of information necessary for restoration
How is a Shortcut Suggested?
NSUserActivity

requiredUserInfoKeys
- Minimal amount of information necessary for restoration
- Used to find patterns
How is a Shortcut Suggested?

NSUserActivity
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>NSUserActivity</th>
<th>&quot;tomato&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>79.0</td>
<td></td>
</tr>
</tbody>
</table>
How is a Shortcut Suggested?
NSUserActivity

NSUserActivity
“tomato”
1
How is a Shortcut Suggested?

NSUserActivity

![Diagram showing two NSUserActivities with the label "tomato" and the number 1]
How is a Shortcut Suggested?

NSUserActivity

<table>
<thead>
<tr>
<th>NSUserActivity</th>
<th>“tomato”</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSUserActivity</td>
<td>“tomato”</td>
<td>1</td>
</tr>
<tr>
<td>NSUserActivity</td>
<td>“tomato”</td>
<td>1</td>
</tr>
</tbody>
</table>

Time
How is a Shortcut Suggested?

NSUserActivity

NSUserActivity
“tomato” 1

NSUserActivity
“tomato” 1

NSUserActivity
“tomato” 1

...
How is a Shortcut Suggested?

NSUserActivity

- NSUserActivity
  - "tomato"
  - 1

- NSUserActivity
  - "tomato"
  - 1

- NSUserActivity
  - "tomato"
  - 1

- NSUserActivity
  - "tomato"
  - 1

- ...
How is a Shortcut Suggested?

Intents

Shortcut Types

<table>
<thead>
<tr>
<th>Parameter Combination</th>
<th>Title</th>
<th>Subtitle</th>
<th>Background</th>
</tr>
</thead>
<tbody>
<tr>
<td>soup</td>
<td>Order</td>
<td>quantity</td>
<td>soup</td>
</tr>
</tbody>
</table>
How is a Shortcut Suggested?

Intents

OrderSoup

soup

quantity

options
How is a Shortcut Suggested?

Intents

OrderSoup

- soup
- quantity
- options

Shortcut Types

OrderSoup

- soup
- quantity
- options

OrderSoup

- soup
- options

OrderSoup

- soup
- quantity
- options
How is a Shortcut Suggested?

Intents
How is a Shortcut Suggested?

Intents

- OrderSoup
  - “tomato”
  - 1
  - ["croutons"]
How is a Shortcut Suggested?

Intents

OrderSoup
“tomato”
1

OrderSoup
“tomato”

[“croutons”]

OrderSoup
“tomato”
1

[“croutons”]

Monday, 11:57 AM
How is a Shortcut Suggested?

Intents

- OrderSoup
  - "tomato"
  - 1
- OrderSoup
  - "tomato"
  - 1
  - ["red pepper"]
- OrderSoup
  - "tomato"
- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
How is a Shortcut Suggested?

Intents

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

Time

Monday, 11:57 AM

Monday, 6:06 PM
How is a Shortcut Suggested?

**Intents**

- **Monday, 11:57 AM**
  - OrderSoup
    - "tomato"
    - 1
    - ["croutons"]

- **Monday, 6:06 PM**
  - OrderSoup
    - "tomato"
    - 1
    - ["red pepper"]

- **Tuesday, 11:59 AM**
  - OrderSoup
    - "tomato"
    - 1
    - ["croutons"]
    - ["red pepper"]
How is a Shortcut Suggested?

Intents

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1
How is a Shortcut Suggested?

Intents

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“tomato”
1

OrderSoup
“croutons”

OrderSoup
“red pepper”

OrderSoup
“croutons”

OrderSoup
“red pepper”

OrderSoup
“croutons”

Shortcut Suggestion
How is a Shortcut Suggested?

**Intents**

- **OrderSoup**
  
  - “tomato”
  
  - 1
  
  - [“croutons”]
  
- **OrderSoup**
  
  - “tomato”
  
  - 1
  
  - [“red pepper”]
  
- **OrderSoup**
  
  - “tomato”
  
  - 1
  
  - [“croutons”]

...

**Shortcut Suggestion**

- Friday, 12:00 PM
How is a Shortcut Suggested?

Intents

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“red pepper”]

OrderSoup
“tomato”
1
[“croutons”]

OrderSoup
“tomato”
1
[“croutons”]
How is a Shortcut Suggested?

Intents

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
  - Monday, 11:57 AM
- OrderSoup
  - "tomato"
  - 1
  - ["red pepper"]
  - Monday, 6:06 PM
- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
  - Tuesday, 11:59 AM
- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
  - Friday, 12:00 PM

...
How is a Shortcut Suggested?

Intents

- OrderSoup
  - “tomato”
  - 1
  - [“croutons”]

- OrderSoup
  - “tomato”
  - 1
  - [“red pepper”]

- OrderSoup
  - “tomato”
  - 1
  - [“croutons”]

- OrderSoup
  - “tomato”
  - 1
  - [“red pepper”]

- OrderSoup
  - “tomato”
  - 1
  - [“croutons”]

- OrderSoup
  - “tomato”
  - 1
  - [“red pepper”]

- OrderSoup
  - “tomato”
  - 1
  - [“croutons”]

- OrderSoup
  - “tomato”
  - 1
How is a Shortcut Suggested?

Intents

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["red pepper"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["red pepper"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]

- OrderSoup
  - "tomato"
  - 1
  - ["croutons"]
What Makes a Good Donation?

Likely to be repeated
What Makes a Good Donation?

Likely to be repeated

Ensure the payload you’re donating is consistent across donations
What Makes a Good Donation?

Likely to be repeated

Ensure the payload you’re donating is consistent across donations

Don’t include timestamps
What Makes a Good Donation?

“Show my meetings for June 5th”

“Show my meetings for today”
What Makes a Good Donation?

Likely to be repeated

Ensure the payload you’re donating is consistent across donations

Don’t include timestamps

Donate once and only once for each user action
Custom Intents

Parameters

Use enums when the values for a parameter are clearly bounded
Custom Intents
Parameters

Use enums when the values for a parameter are clearly bounded

New Localization Workflows with Xcode 10  Executive Ballroom  Wednesday 10:00AM
Custom Intents

Parameters

“Custom” results in an INObject
Custom Intents
Parameters

“Custom” results in an INObject

• Combines an identifier with a display string
Custom Intents

Parameters

“Custom” results in an INObject

- Combines an identifier with a display string
- Identifier can reference an internal object
Custom Intents
Parameters

“Custom” results in an INObject:

- Combines an identifier with a display string
- Identifier can reference an internal object
- Display string conveys that object to the user
Custom Intents
Parameters

“Custom” results in an **INObject**

• Combines an identifier with a display string
• Identifier can reference an internal object
• Display string conveys that object to the user
• Prevents possible implicit dependencies between parameters
Custom Intents

Parameters

```swift
public class MyIntent: INIntent {
    public var objectIdentifier: String?
    public var objectString: String?
}
```
public class MyIntent: INIntent {
    public var objectIdentifier: String?
    public var objectString: String?
}

public class MyIntent: INIntent {
    public var object: INObject?
}
public class MyIntent: INIntent {
    public var objectIdentifier: String?
    public var objectString: String?
}

public class MyIntent: INIntent {
    public var object: INObject?
}
Shortcut Display
Titles, subtitles and images

Understandable

Represent what will happen
Shortcut Testing
Shortcut Testing

Developer settings on device
Shortcut Testing

Developer settings on device

Set up a shortcut with Siri and edit the Xcode scheme to test
Shortcut Testing

Developer settings on device

Set up a shortcut with Siri and edit the Xcode scheme to test
Shortcut Testing

Developer settings on device

Set up a shortcut with Siri and edit the Xcode scheme to test

Create a custom shortcut in the Shortcuts app
Order soup of the day
Send Message
Recipients Ari Weinstein,

It's soup time!

Soup Chef
Order clam chowder
To One Apple Park Way
Order to One Apple Park Way
Start a new order
Order tomato soup
To 123 Soup Street
Order to 123 Soup Street
Start a new order
Privacy
Deleting Donations

User expectations

Prevent suggesting content that’s no longer relevant
Deleting Donations

NSUserActivity

Deleting user activities with corresponding Spotlight items
Deleting user activities with corresponding Spotlight items

```swift
userActivity.contentAttributeSet?.relatedUniqueIdentifier = "my-identifier"

let searchableIndex = CSSearchableIndex.default()
searchableIndex.deleteSearchableItems(withIdentifiers: ["my-identifier"]) { (error) in
```
Deleting Donations

NSUserActivity

Deleting user activities with corresponding Spotlight items

```swift
userActivity.contentAttributeSet?.relatedUniqueIdentifier = "my-identifier"

let searchableIndex = CSSearchableIndex.default()
searchableIndex.deleteSearchableItems(withIdentifiers: ["my-identifier"]) { (error) in 
}
```
Deleting Donations

NSUserActivity

Persistent identifier property on NSUserActivity

Use the identifier to later delete specific NSUserActivities
// Deleting NSUserActivity using persistent identifier

let userActivity = NSUserActivity(activityType: "my-activity-type")
// Configure user activity...

let persistentIdentifier = NSUserActivityPersistentIdentifier("persistent-identifier")
userActivity.persistentIdentifier = persistentIdentifier

NSUserActivity.deleteSavedUserActivities(withPersistentIdentifiers: [persistentIdentifier]) {
}

NSUserActivity.deleteAllSavedUserActivities {
}
// Deleting NSUserActivity using persistent identifier

let userActivity = NSUserActivity(activityType: "my-activity-type")
// Configure user activity...

let persistentIdentifier = NSUserActivityPersistentIdentifier("persistent-identifier")
userActivity.persistentIdentifier = persistentIdentifier

NSUserActivity.deleteSavedUserActivities(withPersistentIdentifiers: [persistentIdentifier]) {
}

NSUserActivity.deleteAllSavedUserActivities {
}
// Deleting NSUserActivity using persistent identifier

let userActivity = NSUserActivity(activityType: "my-activity-type")
// Configure user activity...

let persistentIdentifier = NSUserActivityPersistentIdentifier("persistent-identifier")
userActivity.persistentIdentifier = persistentIdentifier

NSUserActivity.deleteSavedUserActivities(withPersistentIdentifiers: [persistentIdentifier]) {
}

NSUserActivity.deleteAllSavedUserActivities {
}
Deleting Donations

Intents

Identifier and group identifier properties on INInteraction

Use the identifier or group identifier to later delete one or more INInteractions
// Donating
let interaction = INInteraction(intent: intent, response: response)

interaction.identifier = "my-identifier"
interaction.groupIdentifier = "my-group-identifier"

interaction.donate { (error) in }

// Deleting the donation
INInteraction.delete(with: ["my-identifier"]) { (error) in }

INInteraction.delete(with: "my-group-identifier") { (error) in }

INInteraction.deleteAll { (error) in }
// Donating
let interaction = INInteraction(intent: intent, response: response)
interaction.identifier = "my-identifier"
interaction.groupIdentifier = "my-group-identifier"
interaction.donate { (error) in }

// Deleting the donation
INInteraction.delete(with: ["my-identifier"]) { (error) in }

INInteraction.delete(with: "my-group-identifier") { (error) in }

INInteraction.deleteAll { (error) in }
// Donating
let interaction = INInteraction(intent: intent, response: response)
interaction.identifier = "my-identifier"
interaction.groupIdentifier = "my-group-identifier"
interaction.donate { (error) in
}

// Deleting the donation
INInteraction.delete(with: ["my-identifier"]) { (error) in
}

INInteraction.delete(with: "my-group-identifier") { (error) in

INInteraction.deleteAll { (error) in
}
// Donating
let interaction = INInteraction(intent: intent, response: response)
interaction.identifier = "my-identifier"
interaction.groupIdentifier = "my-group-identifier"
interaction.donate { (error) in ...

// Deleting the donation
INInteraction.delete(with: ["my-identifier"]) { (error) in ...

INInteraction.delete(with: "my-group-identifier") { (error) in ...

INInteraction.deleteAll { (error) in ...
Media Shortcuts
Play Media Intent

INPlayMediaIntent
Play Media Intent

INPlayMediaIntent

Launches app in background
Play Media Intent

INPlayMediaIntent

Launches app in background

Suggested in playback controls on Lock Screen
Play Media Intent

INPlayMediaIntent

Launches app in background

Suggested in playback controls on Lock Screen

Works on HomePod
Upcoming Media

Use INUpcomingMedia API to tell the system about new media the user hasn’t yet consumed.

Great for podcasts, TV shows.
Summary

Enables powerful new experiences

Provides new ways to engage users

Adopt with NSUserActivity or Intents

See “Building for Voice with Siri Shortcuts”
More Information

https://developer.apple.com/wwdc18/211

<table>
<thead>
<tr>
<th>Event</th>
<th>Location</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Building for Voice with Siri Shortcuts</td>
<td>Hall 2</td>
<td>Wednesday 10:00AM</td>
</tr>
<tr>
<td>Siri Shortcuts Lab</td>
<td>Technology Lab 7</td>
<td>Wednesday 12:00PM</td>
</tr>
<tr>
<td>Siri Shortcuts Lab</td>
<td>Technology Lab 11</td>
<td>Thursday 11:00AM</td>
</tr>
</tbody>
</table>