

# Building Apps with ResearchKit

Session 213

John Earl Software Engineer

# Agenda

What is ResearchKit?

Designing an App-based Study

Building Apps with ResearchKit

Contributing to ResearchKit

What is ResearchKit?

# What is ResearchKit?

[github.com/researchkit/researchkit](https://github.com/researchkit/researchkit)

# Medical Research

## MEDICAL RESEARCH Volunteers Needed!

Do you have diabetes?

We are seeking volunteers for a study about diabetes and gluten.  
You must have diabetes, not smoke and be 18-40 years old. Participants will  
be asked to attend a 2 hour focus group or a 40 minute telephone interview.  
To volunteer please contact us at [800-555-0123](tel:800-555-0123) or [info@nri.org](mailto:info@nri.org)

National Research Institute

National Research Institute

National Research Institute

National Research Institute

National Research Institute

National Research Institute

National Research Institute

National Research Institute

National Research Institute

# Medical Research



# Medical Research

Limited participation

Subjective and infrequent data

One-way communication

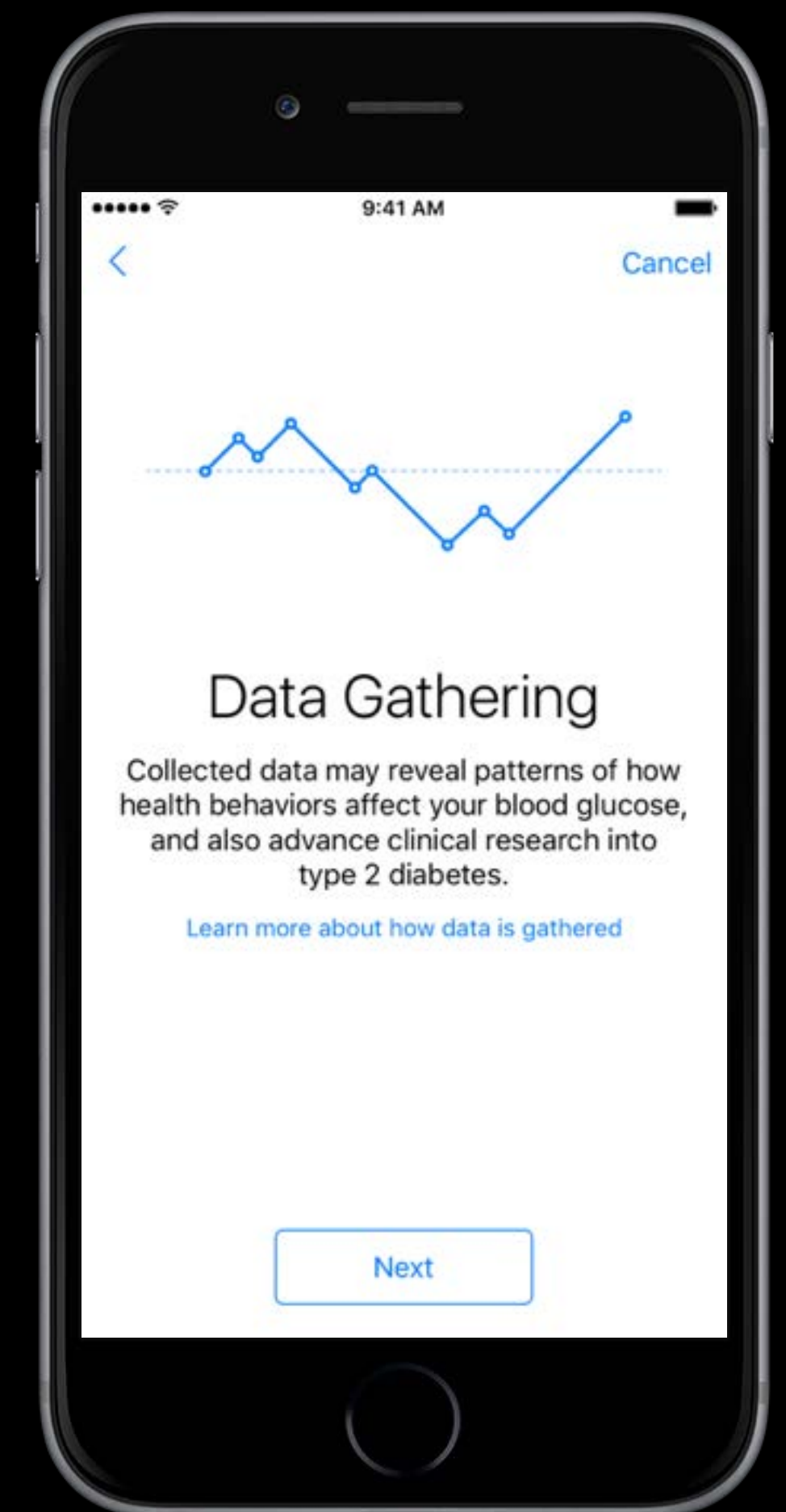


# Research Apps

Limited participation

Subjective and infrequent data

One-way communication

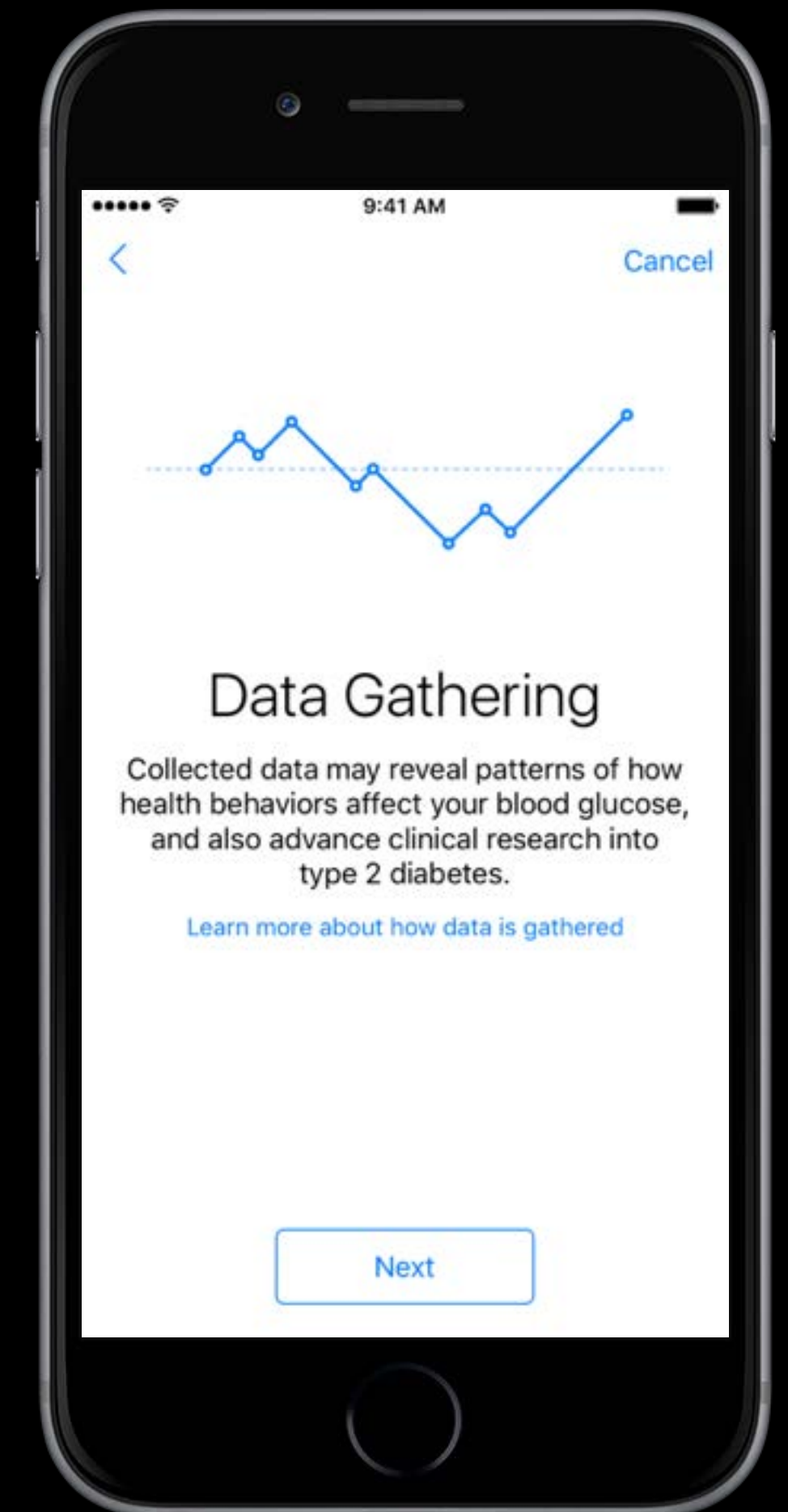


# Research Apps

Limited participation → Wider reach

Subjective and infrequent data

One-way communication

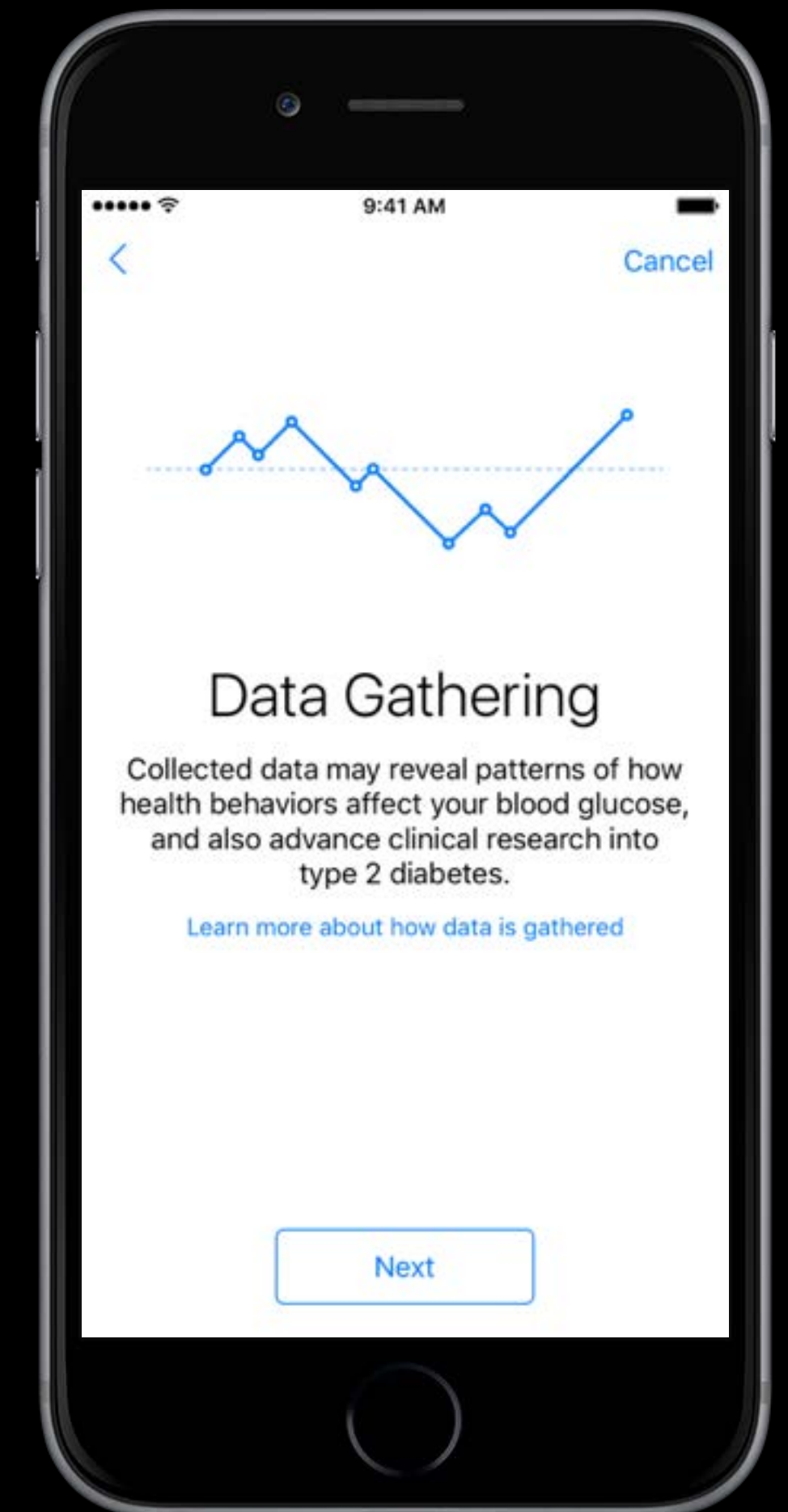


# Research Apps

Limited participation → Wider reach

Subjective and infrequent data → Continuous streaming of data

One-way communication

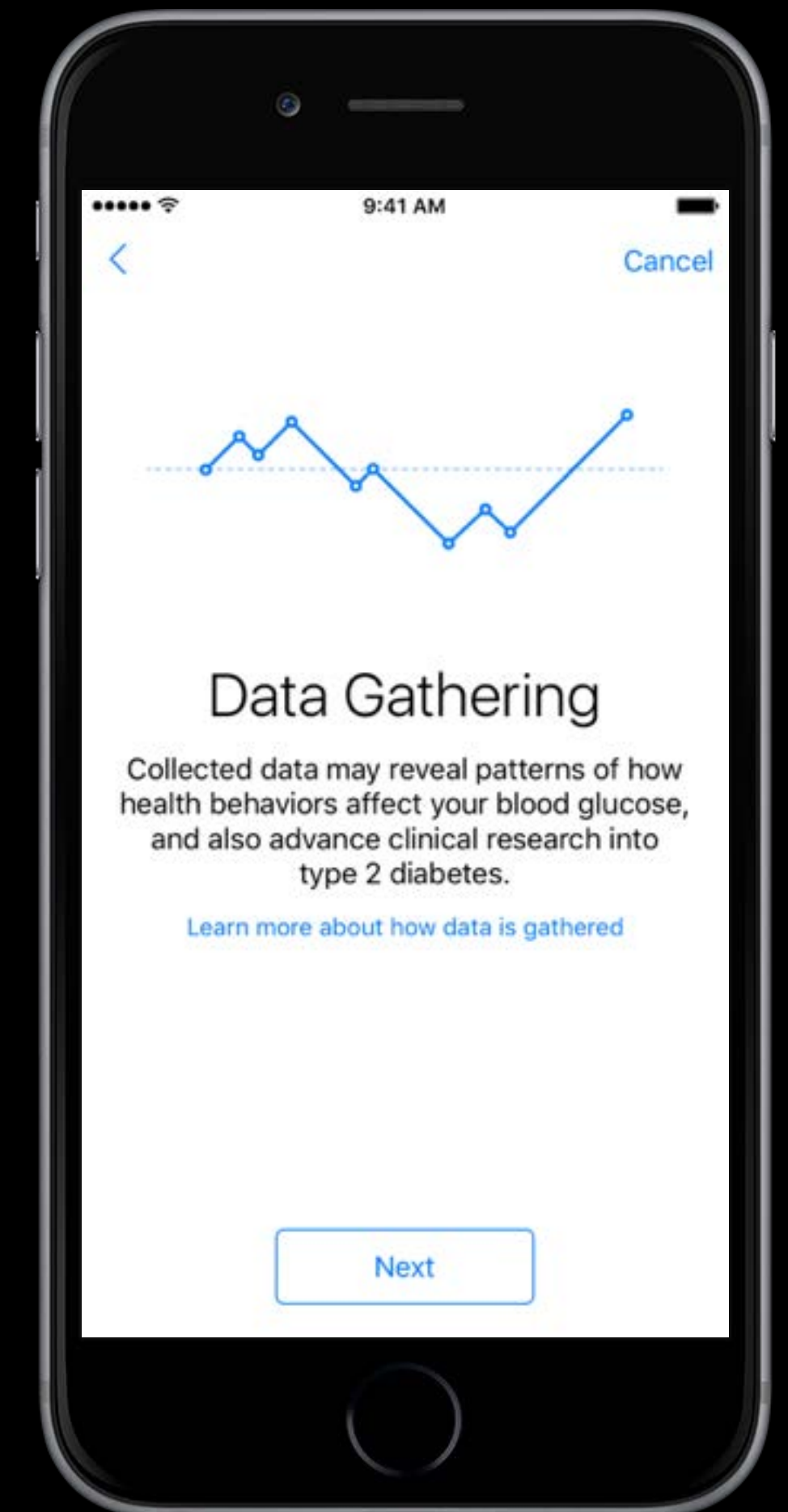


# Research Apps

Limited participation → Wider reach

Subjective and infrequent data → Continuous streaming of data

One-way communication → Feedback for participants



# ResearchKit

Open source framework

Easily create research apps

Extends HealthKit



# Surveys



The image shows a smartphone screen with a survey interface. At the top, the status bar shows signal strength, the time 9:41 AM, and battery level. The survey header includes a back arrow, 'Step 3 of 4', and a 'Cancel' link. The main question is 'What is your blood pressure?'. Below this is a note: 'If you have recently entered a measurement in Health, it will be pre-filled.' The survey content is titled 'BLOOD PRESSURE' and contains two input fields. The first field is labeled 'Systolic' and has the value '115 mm Hg'. The second field is labeled 'Diastolic' and has the value '80 mm Hg'. At the bottom, there are two buttons: 'Next' (highlighted with a blue border) and 'Skip'.

9:41 AM

< Step 3 of 4 Cancel

What is your blood pressure?

If you have recently entered a measurement in Health, it will be pre-filled.

BLOOD PRESSURE

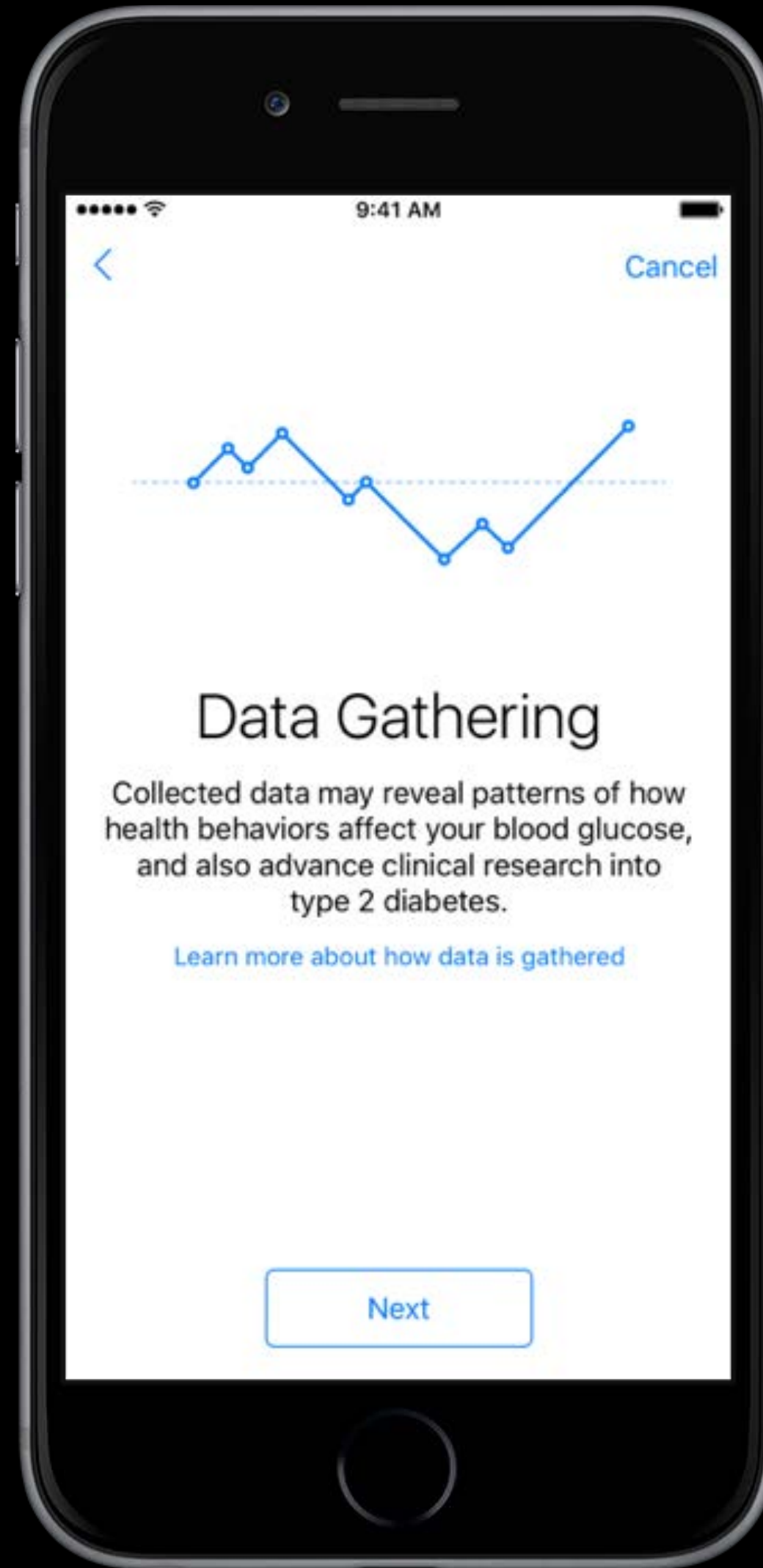
Systolic 115 mm Hg

Diastolic 80 mm Hg

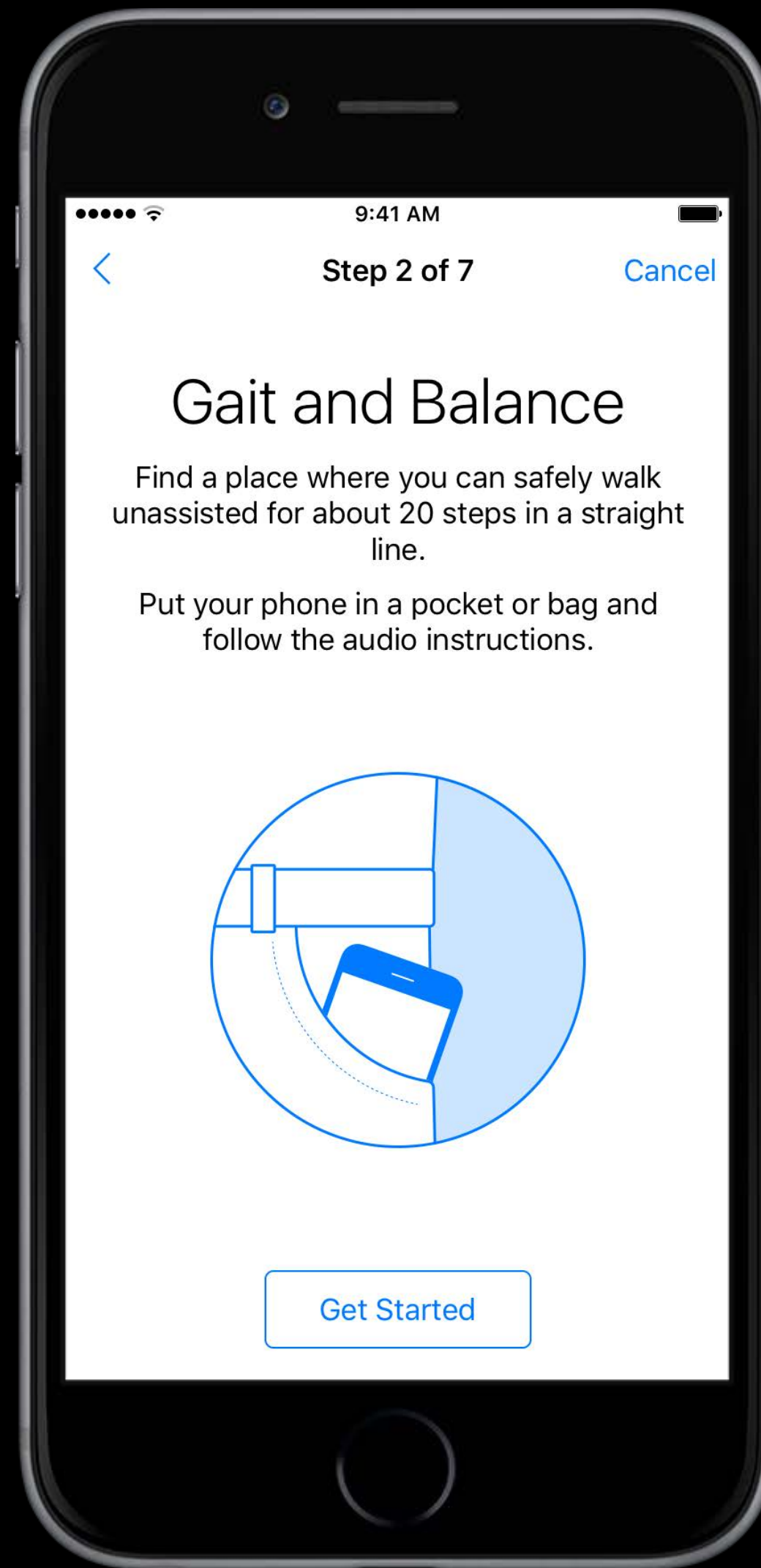
Next

Skip

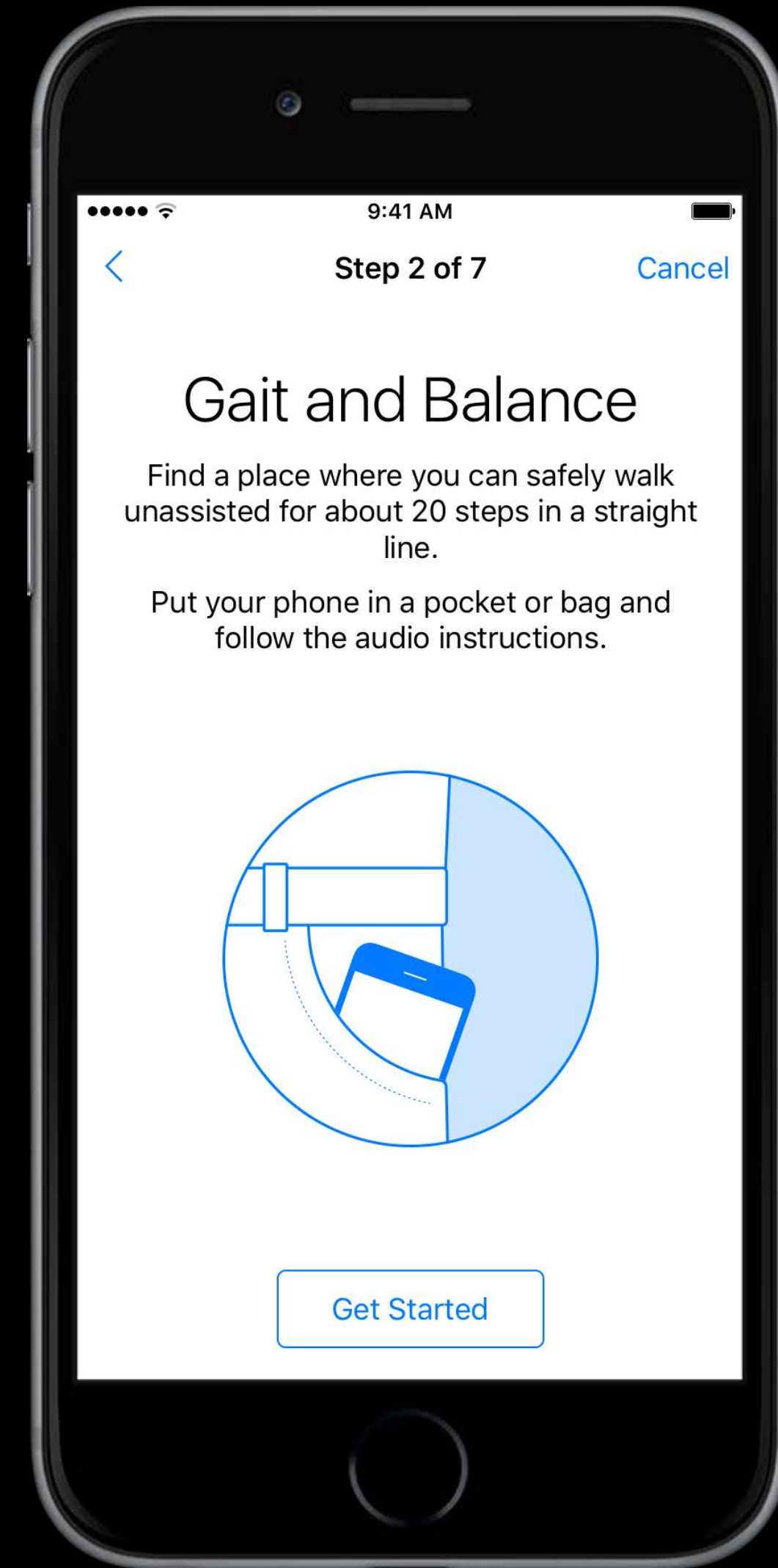
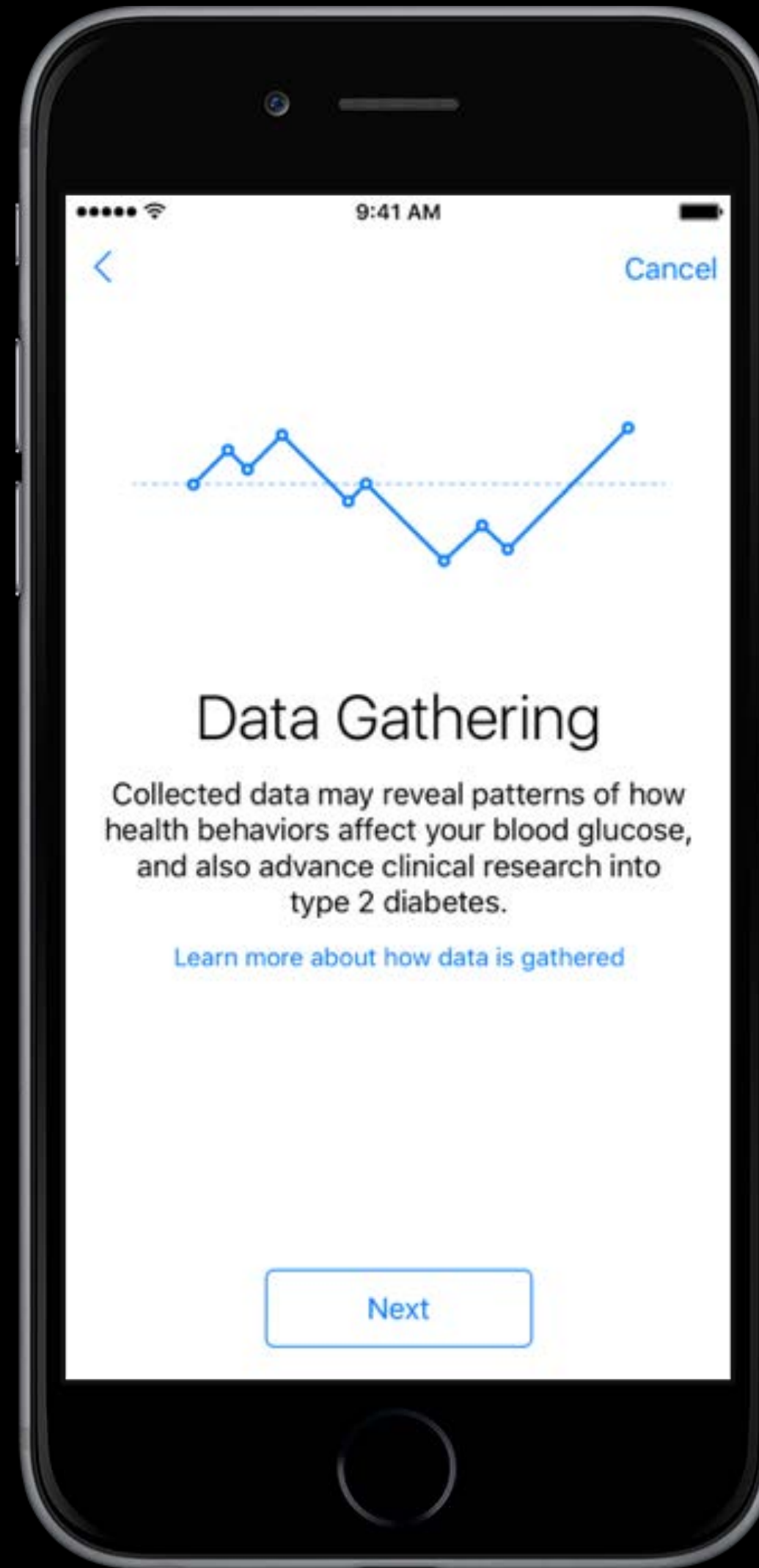
# Consent



# Active Tasks



# ResearchKit Modules



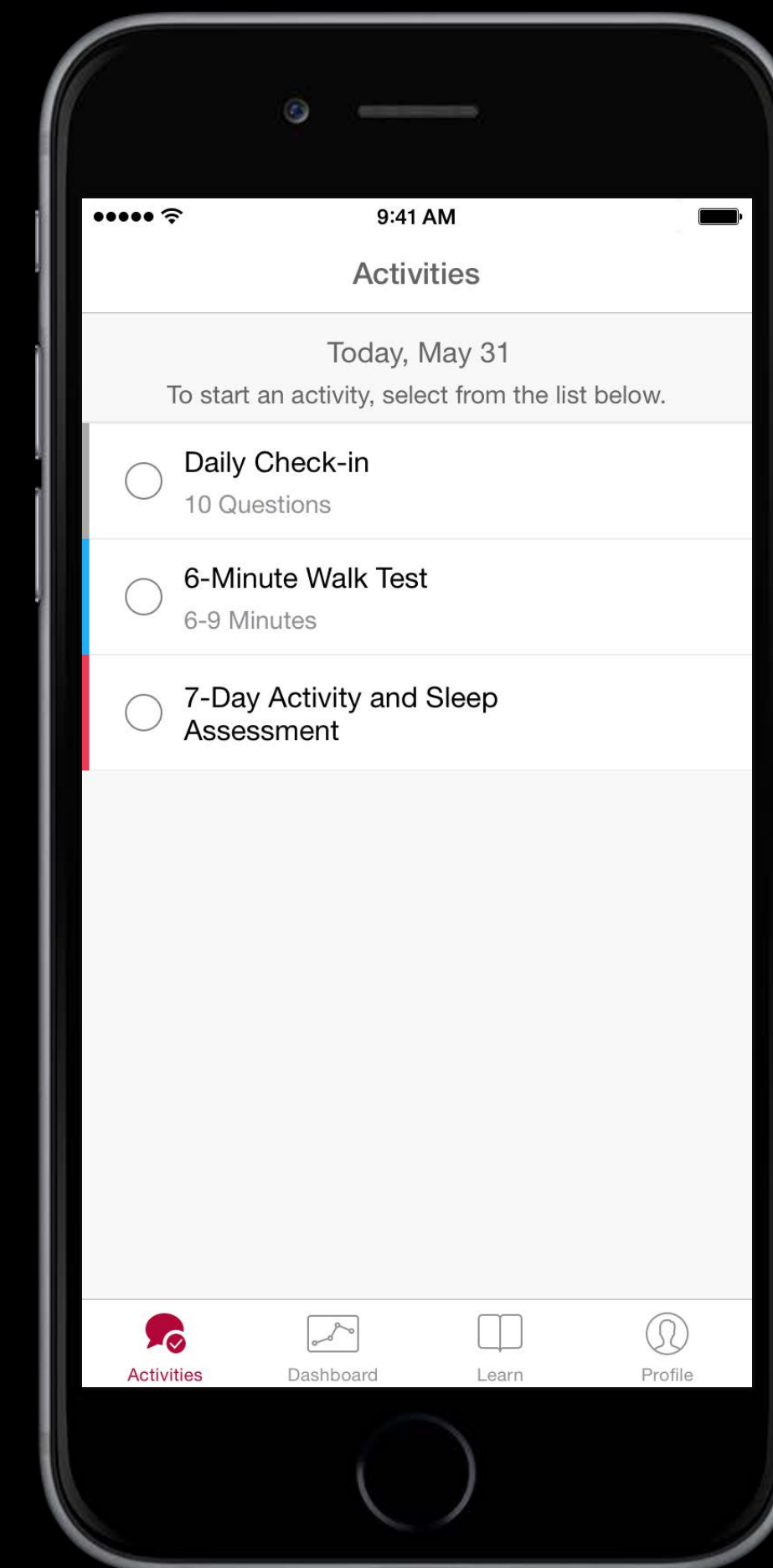
# Apps Using ResearchKit



# Common Themes

## Scheduled activities

- Surveys
- Active tasks



# Common Themes

Scheduled activities

- Surveys
- Active tasks

Passive data collection

- HealthKit
- CoreMotion



# Common Themes

Scheduled activities

- Surveys
- Active tasks

Passive data collection

- HealthKit
- CoreMotion

Dashboard



# Designing an App-based Study







Institutional Review Board or Ethics Committee

Partners HealthCare System  
Research Consent Form

General Template  
Version Date: February 2010

Protocol Title: Smartphone app for management of type 2 diabetes

Principal Investigator: Stanley Shaw, MD PhD

Site Principal Investigator: Stanley Shaw, MD PhD

Description of Subject Population: Individuals 18 or over with type 2 diabetes or pre-diabetes who live in the United States

About this consent form

The following pages explain important information about the Glucosuccess app, and how the app is part of a research study.

Please read this form carefully.

Taking part in this research study is voluntary and is up to you. If you decide to take part in this research study, you must sign this form to show that you want to take part. We will give you a signed copy of this form to keep. If you have any questions about the research or about this form, please ask us.

People who agree to take part in research studies are called “subjects”. This term will be used throughout this consent form. Partners HealthCare System is made up of Partners hospitals (including Massachusetts General Hospital, where this study originates), health care providers, and researchers. In the rest of this consent form, we refer to the Partners system simply as “Partners”.

This study is sponsored by the Center for Assessment Technology and Continuous Health (CATCH) at Massachusetts General Hospital in Boston, MA.

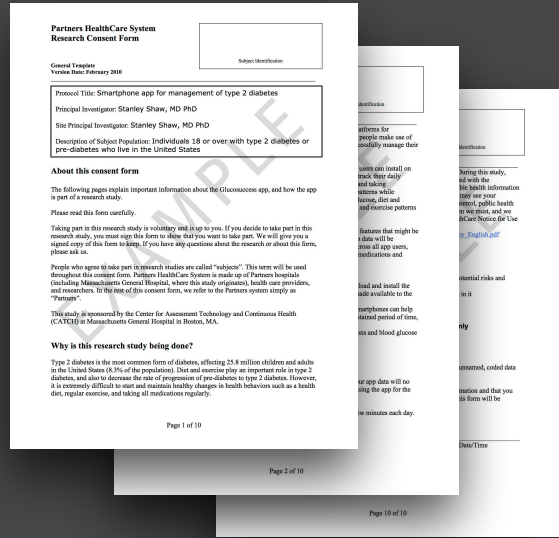
Why is this research study being done?

Type 2 diabetes is the most common form of diabetes, affecting 25.8 million children and adults in the United States (8.3% of the population). Diet and exercise play an important role in type 2 diabetes, and also to decrease the rate of progression of pre-diabetes to type 2 diabetes. However, it is extremely difficult to start and maintain healthy changes in health behaviors such as a health diet, regular exercise, and taking all medications regularly.









# Building Apps with ResearchKit

# Overview

**ORKTask**

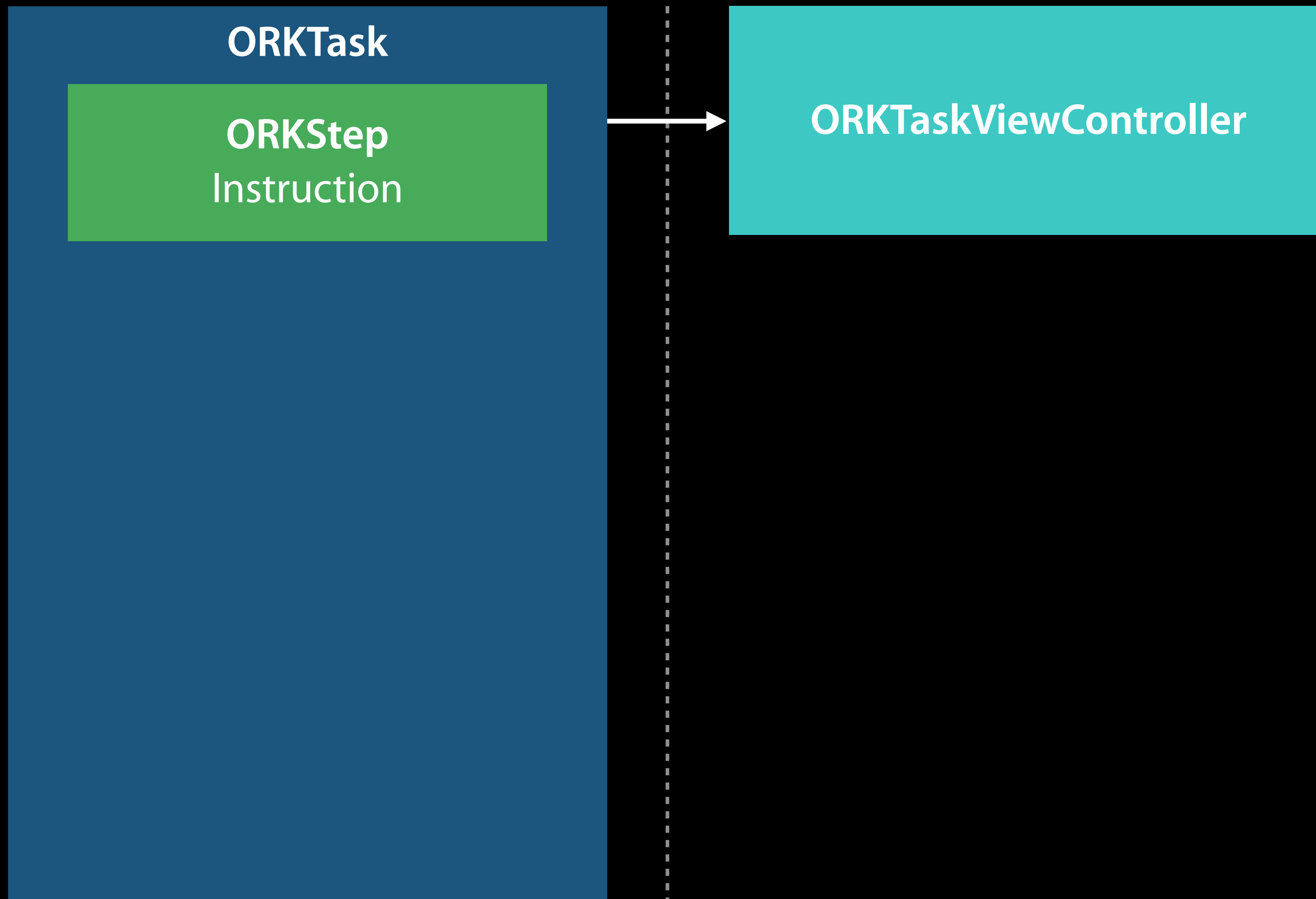


# Overview

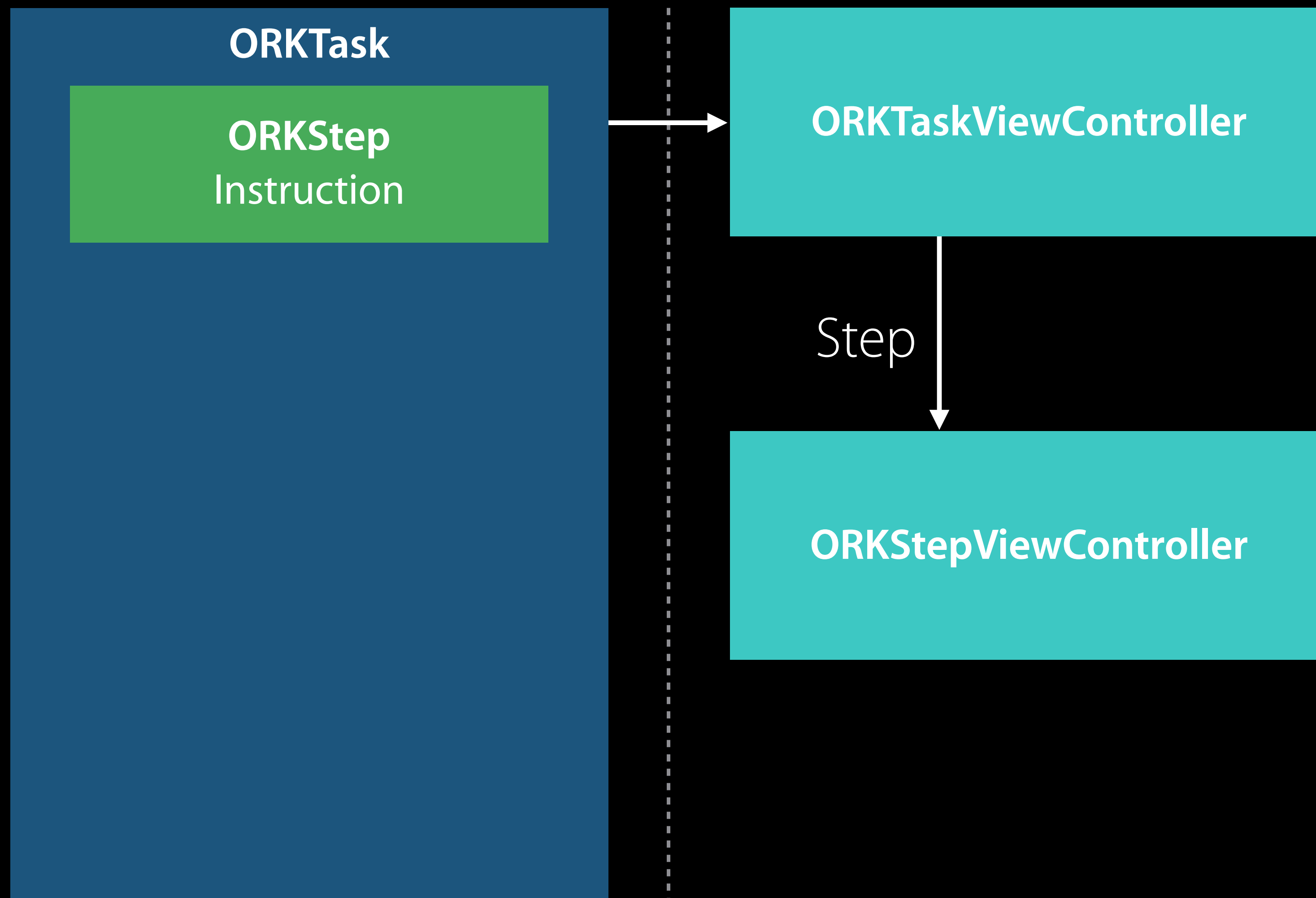
**ORKTask**

**ORKStep**  
Instruction

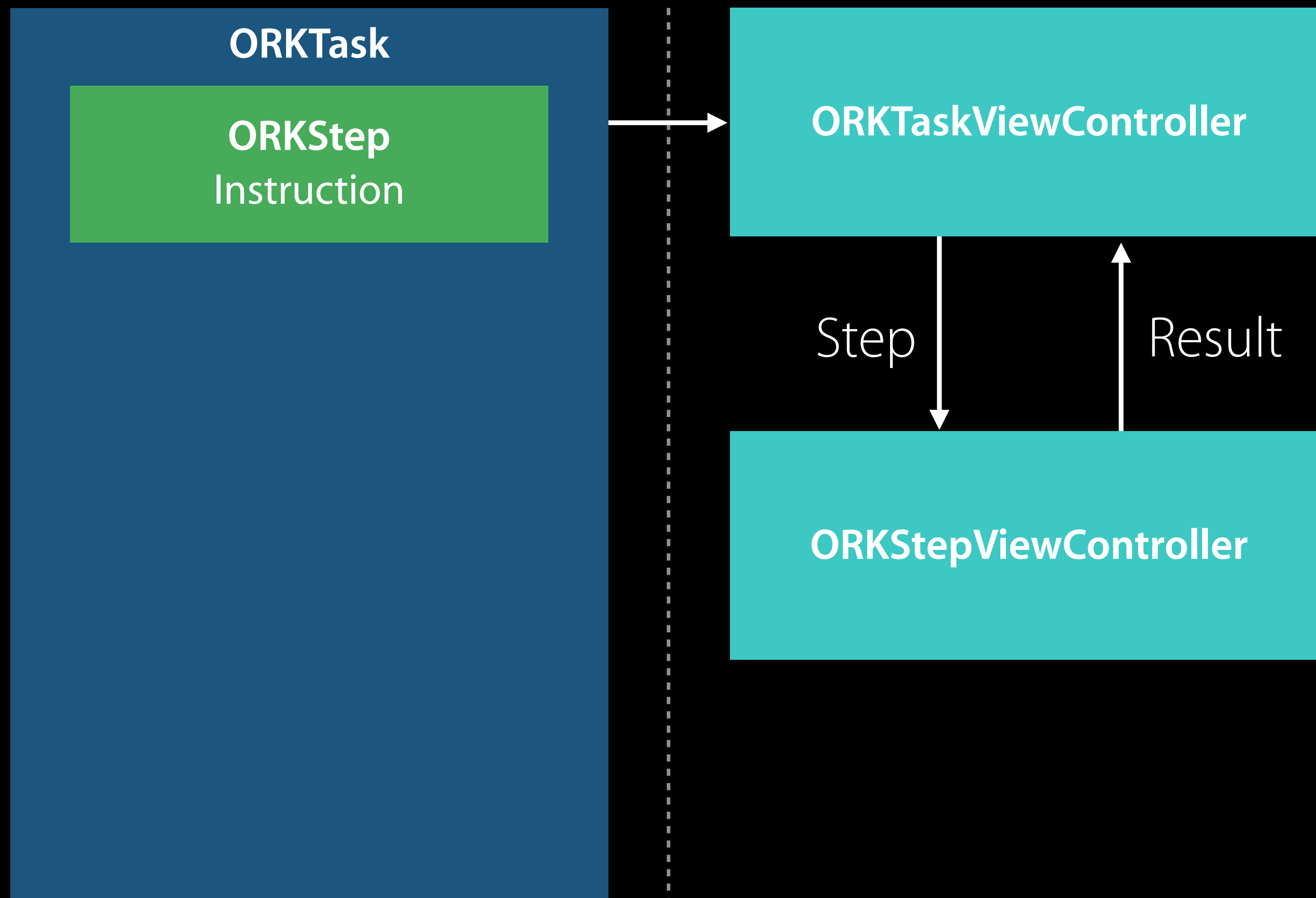
# Overview



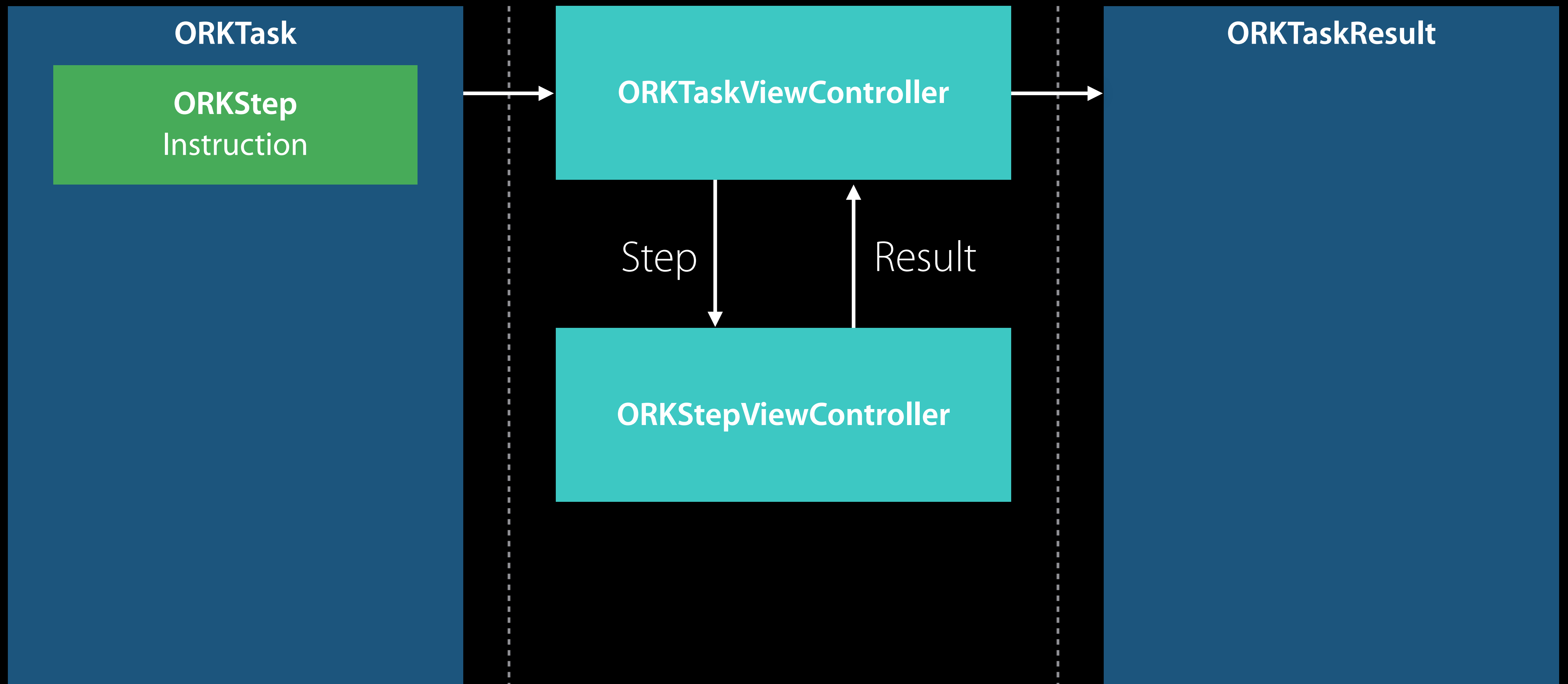
# Overview



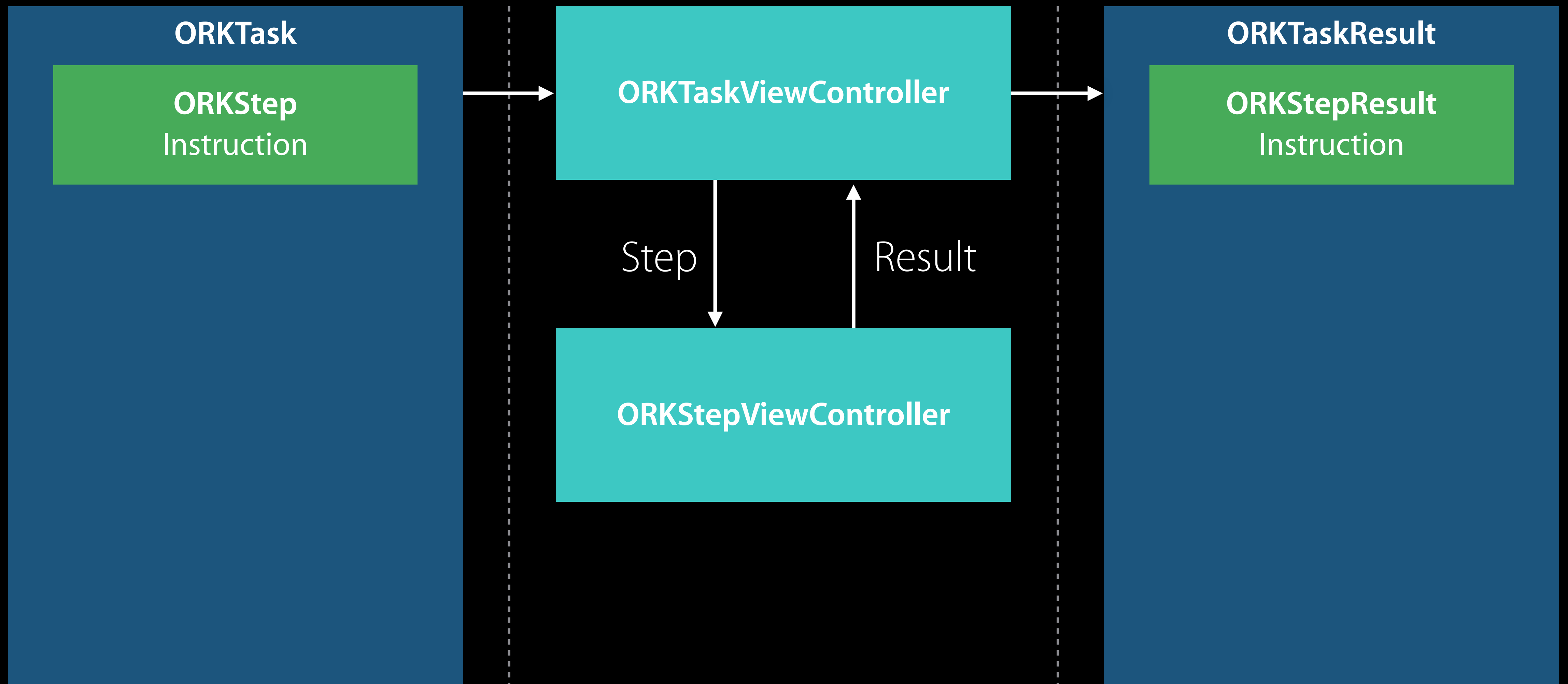
# Overview



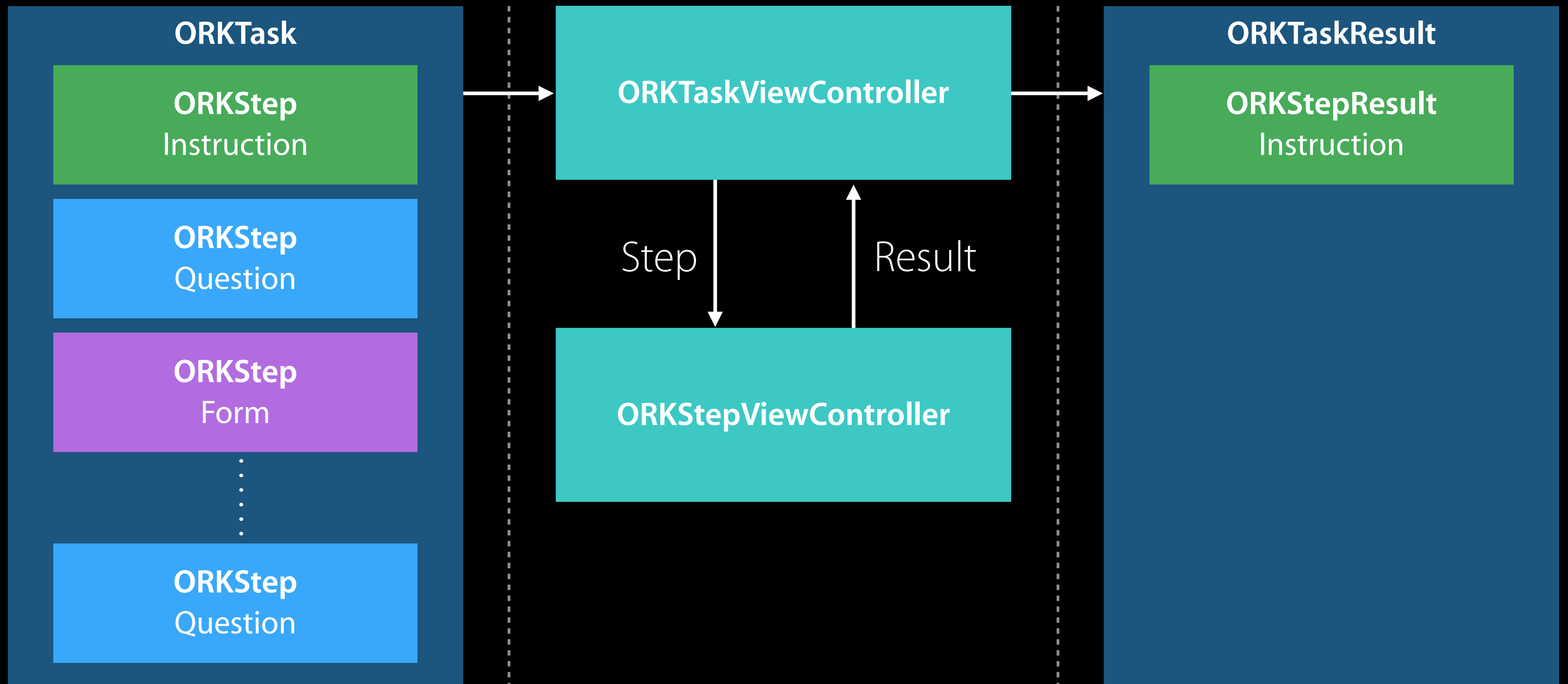
# Overview



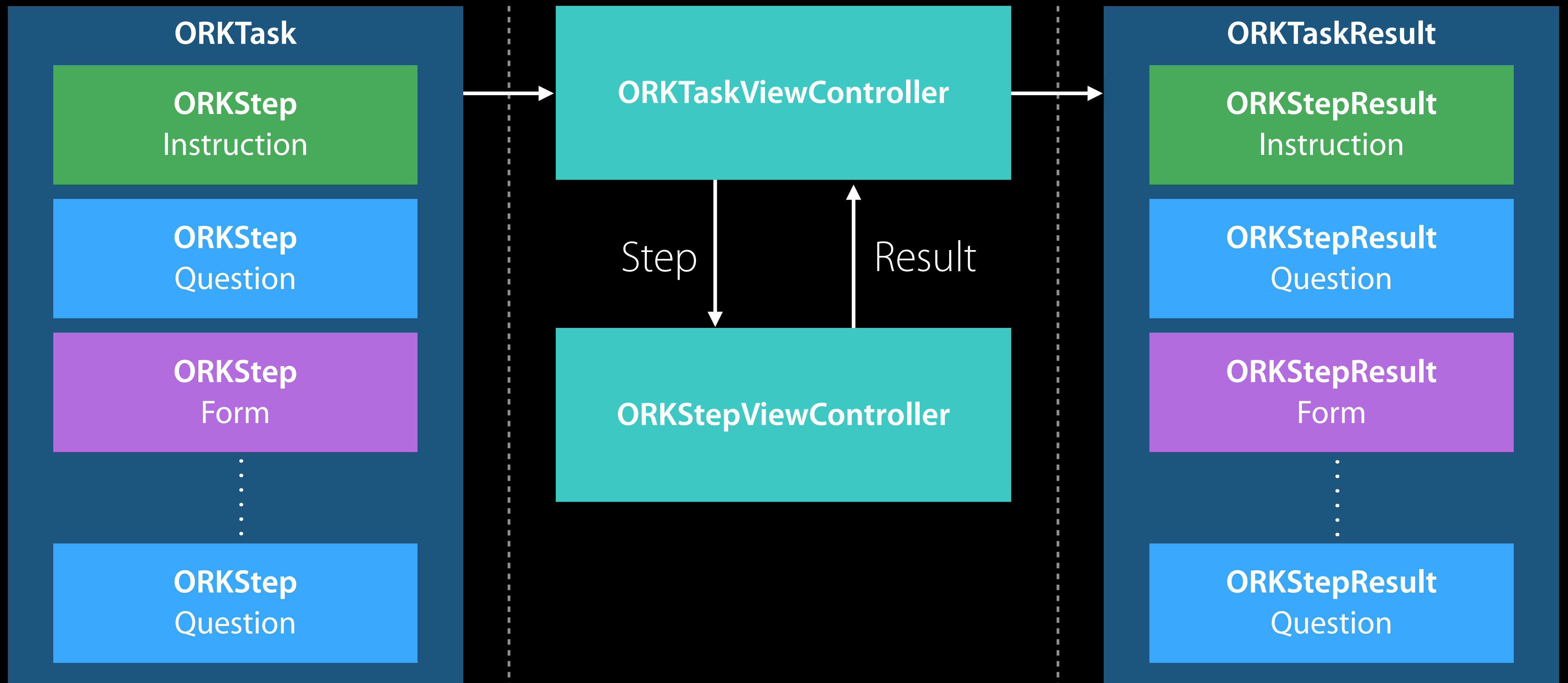
# Overview



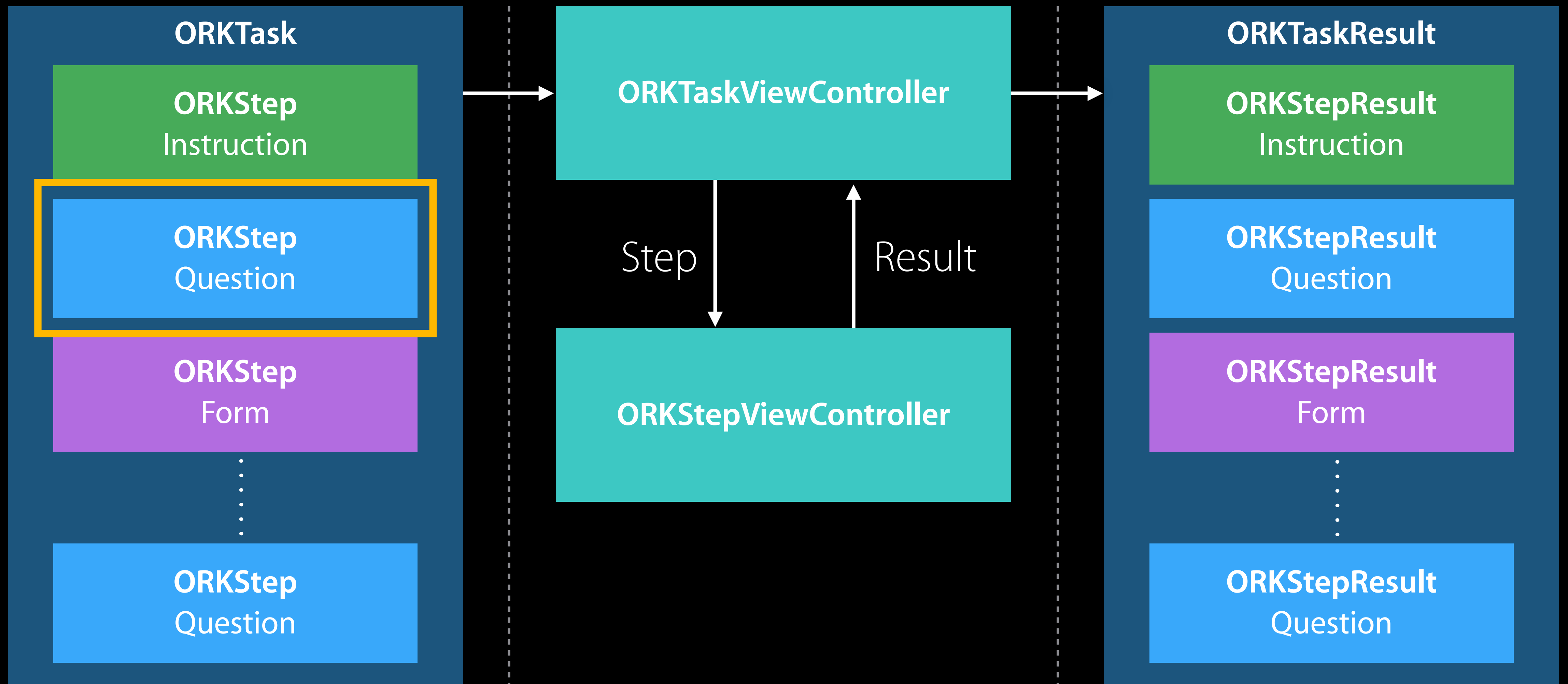
# Overview



# Overview



# Overview

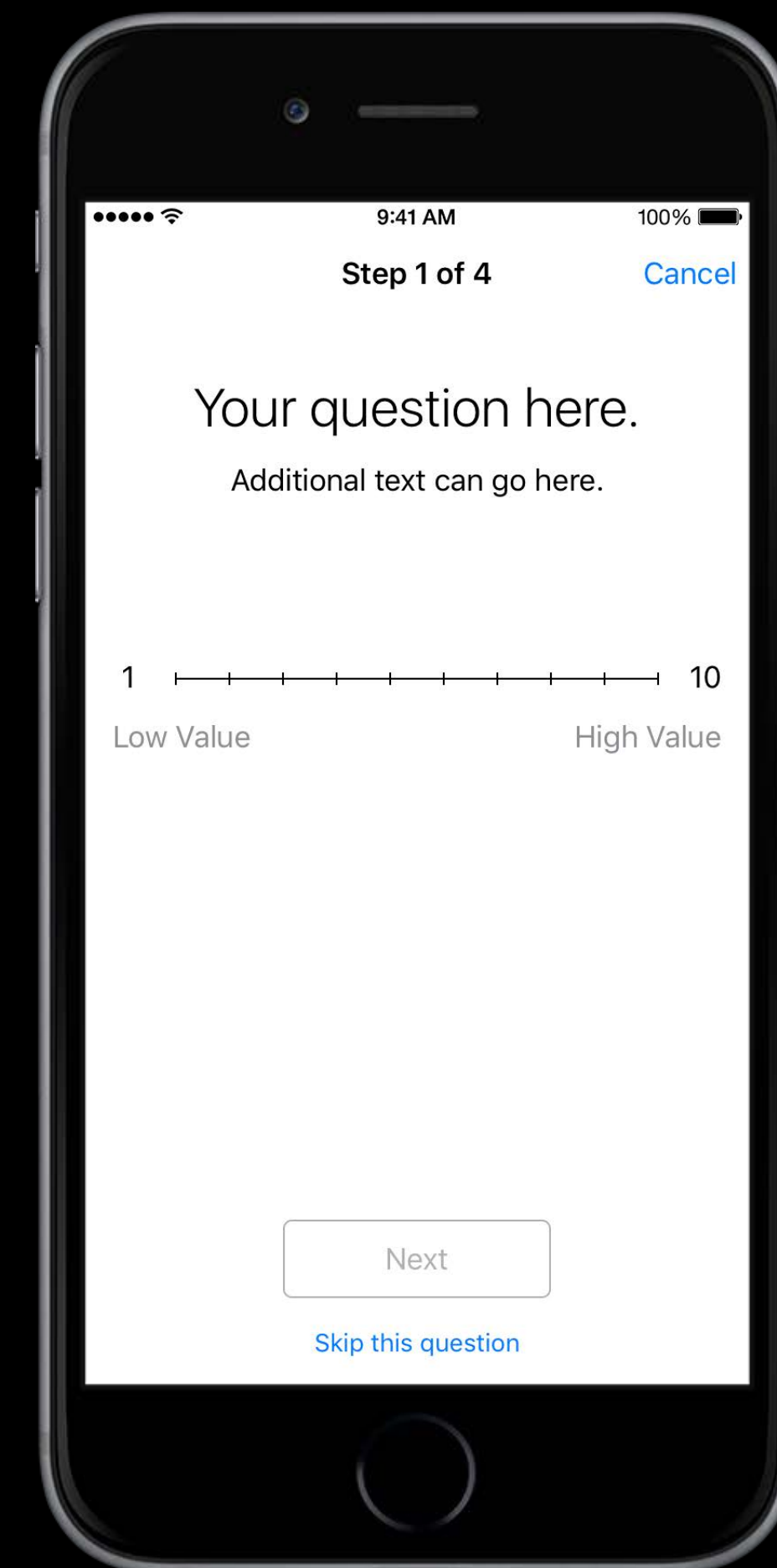


# ORKStep

## The basic template

Used for:

- An instruction
- Survey question
- Multi-question form
- A countdown timer
- A memory game
- ...and so on



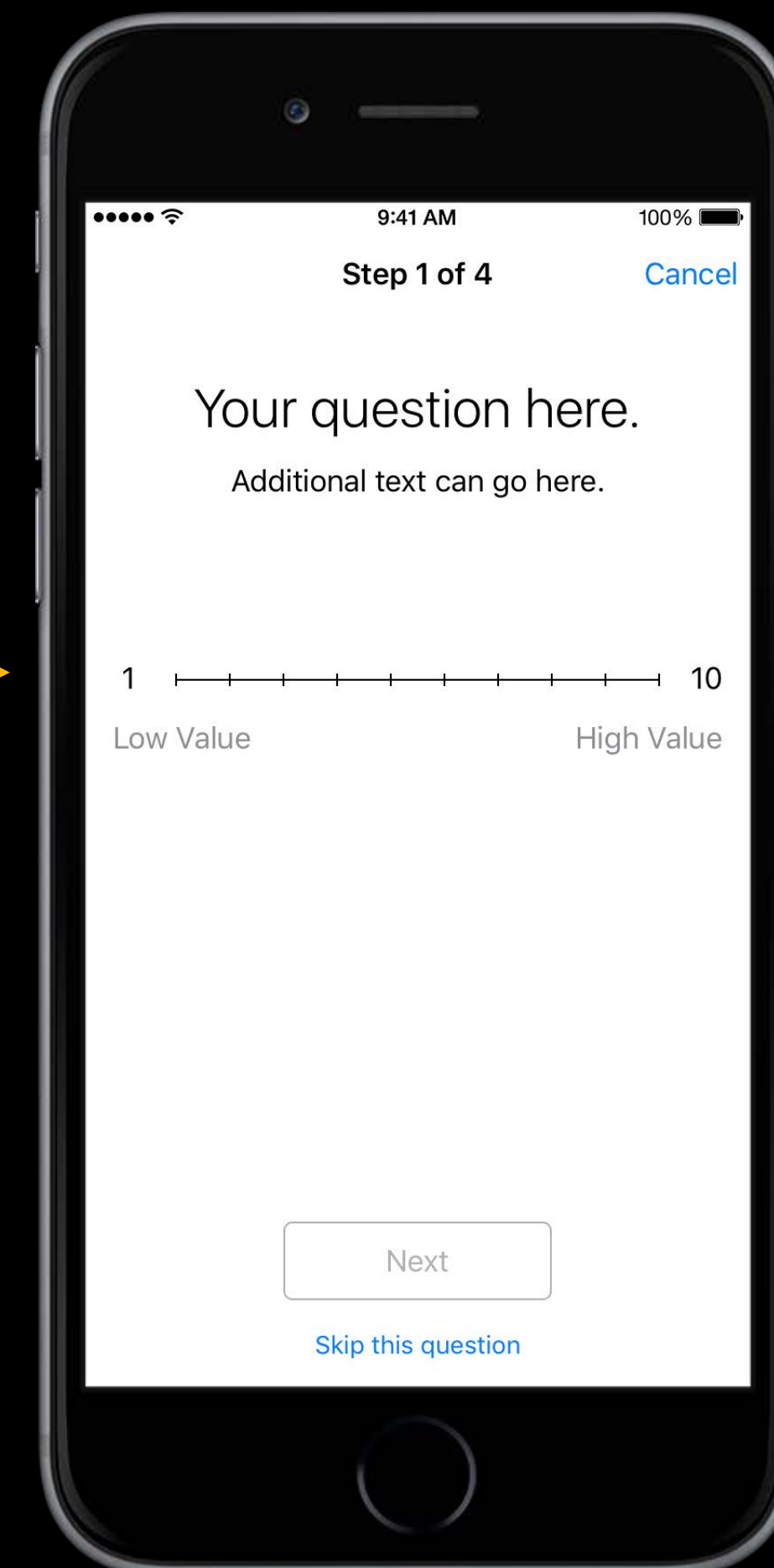
# ORKStep

## The basic template

Used for:

- An instruction
- Survey question
- Multi-question form
- A countdown timer
- A memory game
- ...and so on

Step Content →



# ORKStep

## The basic template

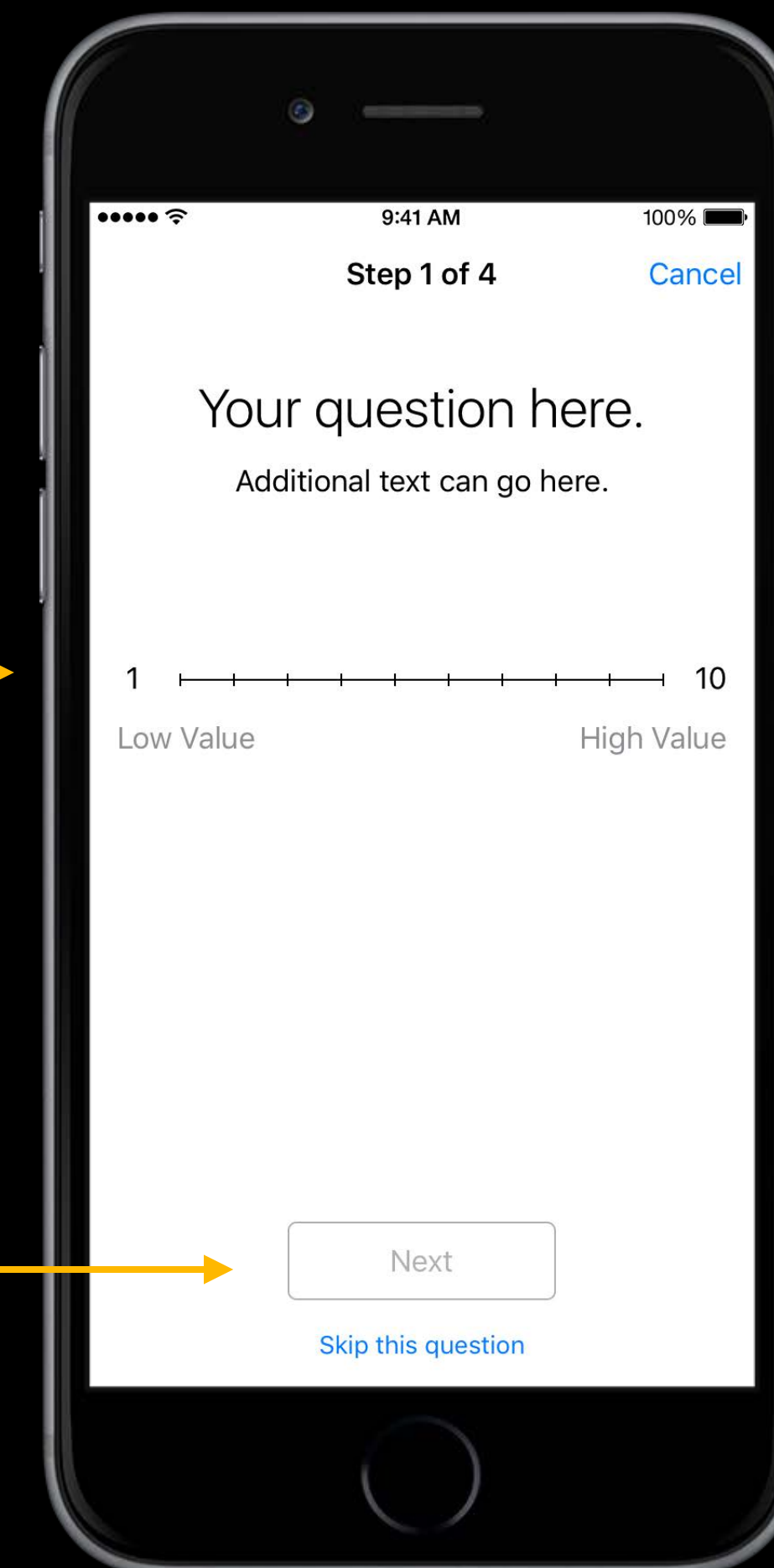
Used for:

- An instruction
- Survey question
- Multi-question form
- A countdown timer
- A memory game
- ...and so on

Step Content



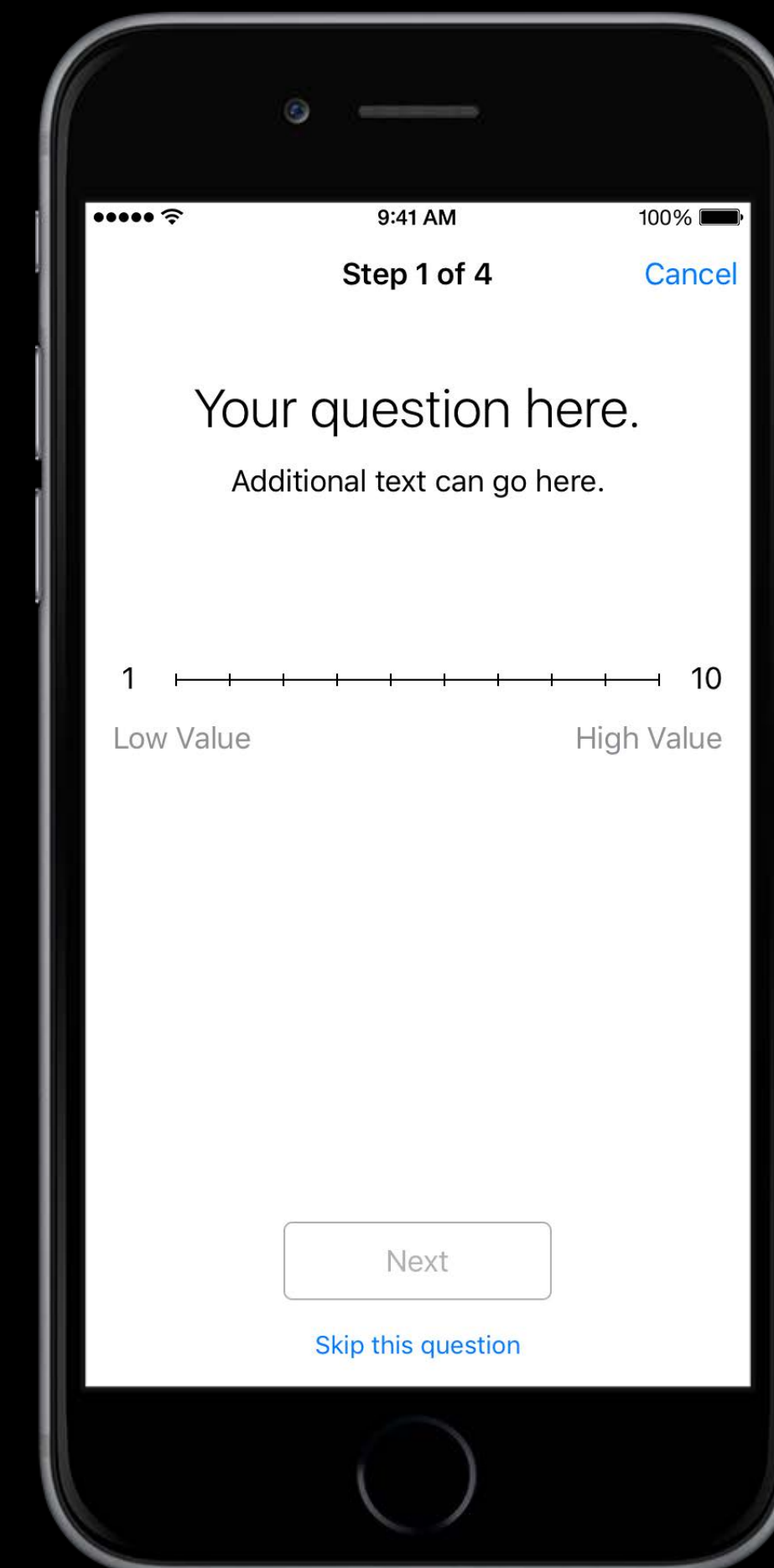
Forward Navigation



# ORKStep

Steps in code

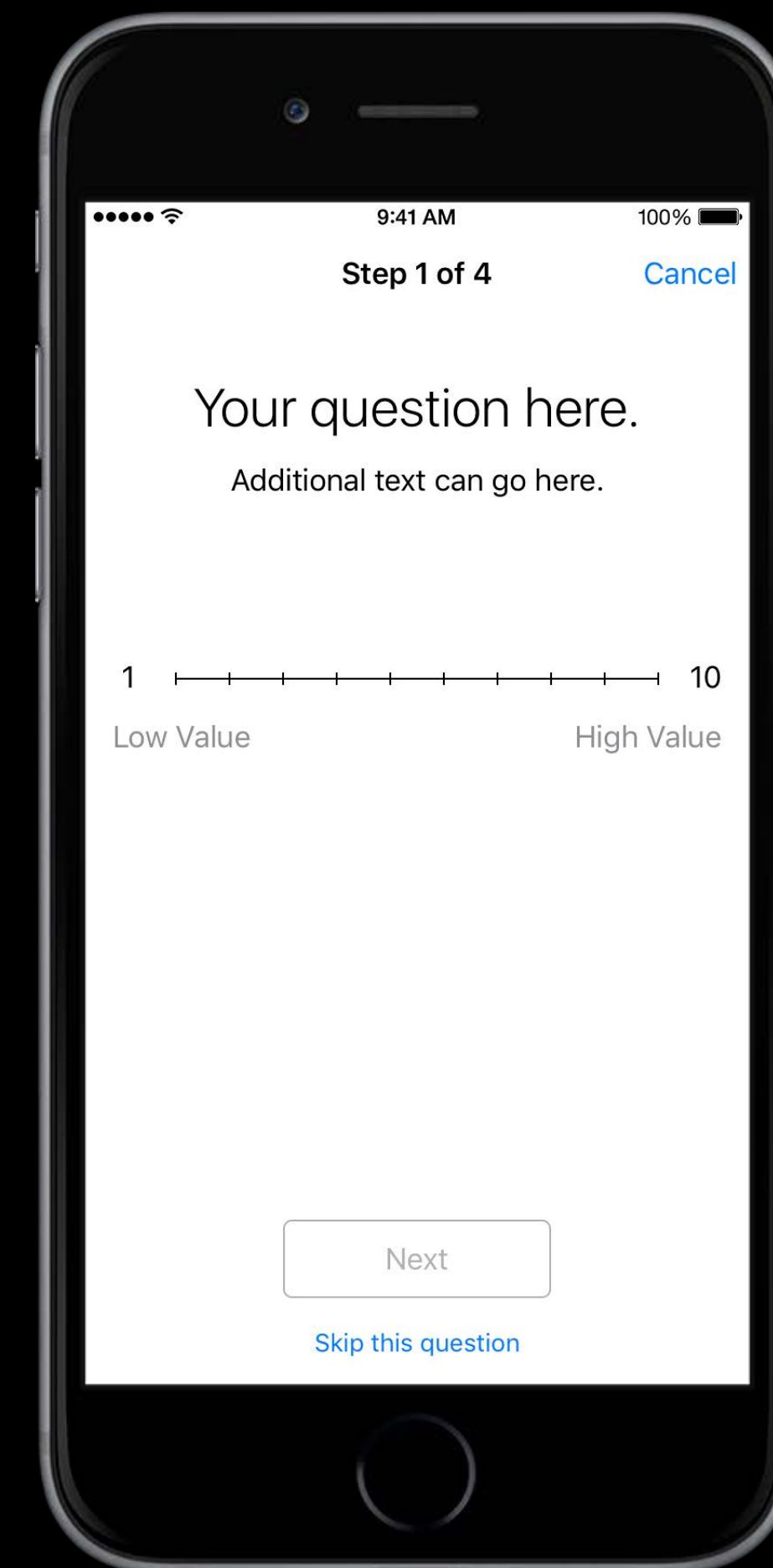
```
class ORKStep : NSObject {
```



# ORKStep

## Steps in code

```
class ORKStep : NSObject {  
  
    var identifier: String { get }  
}
```

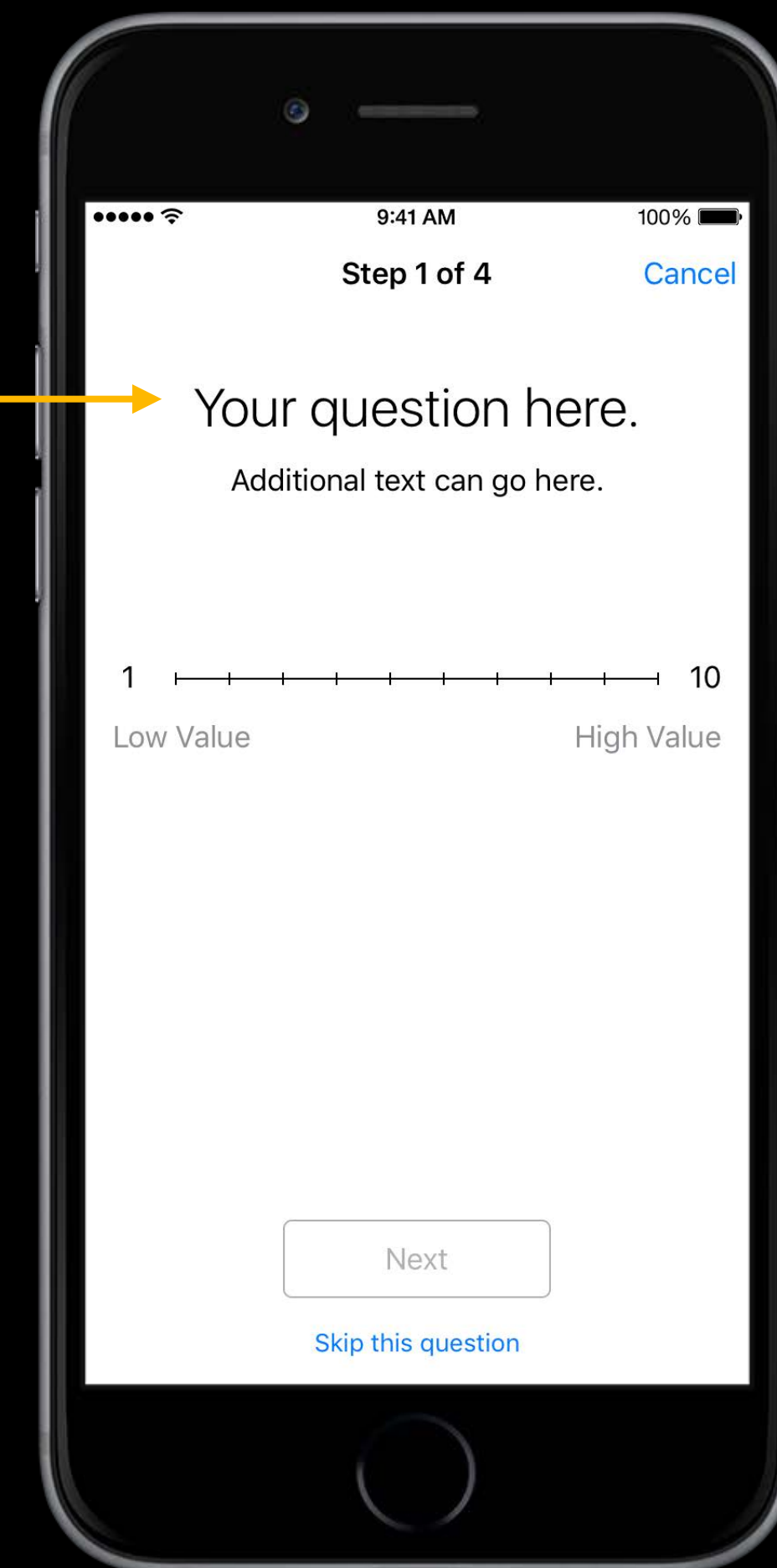


# ORKStep

## Steps in code

```
class ORKStep : NSObject {  
  
    var identifier: String { get }  
  
    var title: String?
```

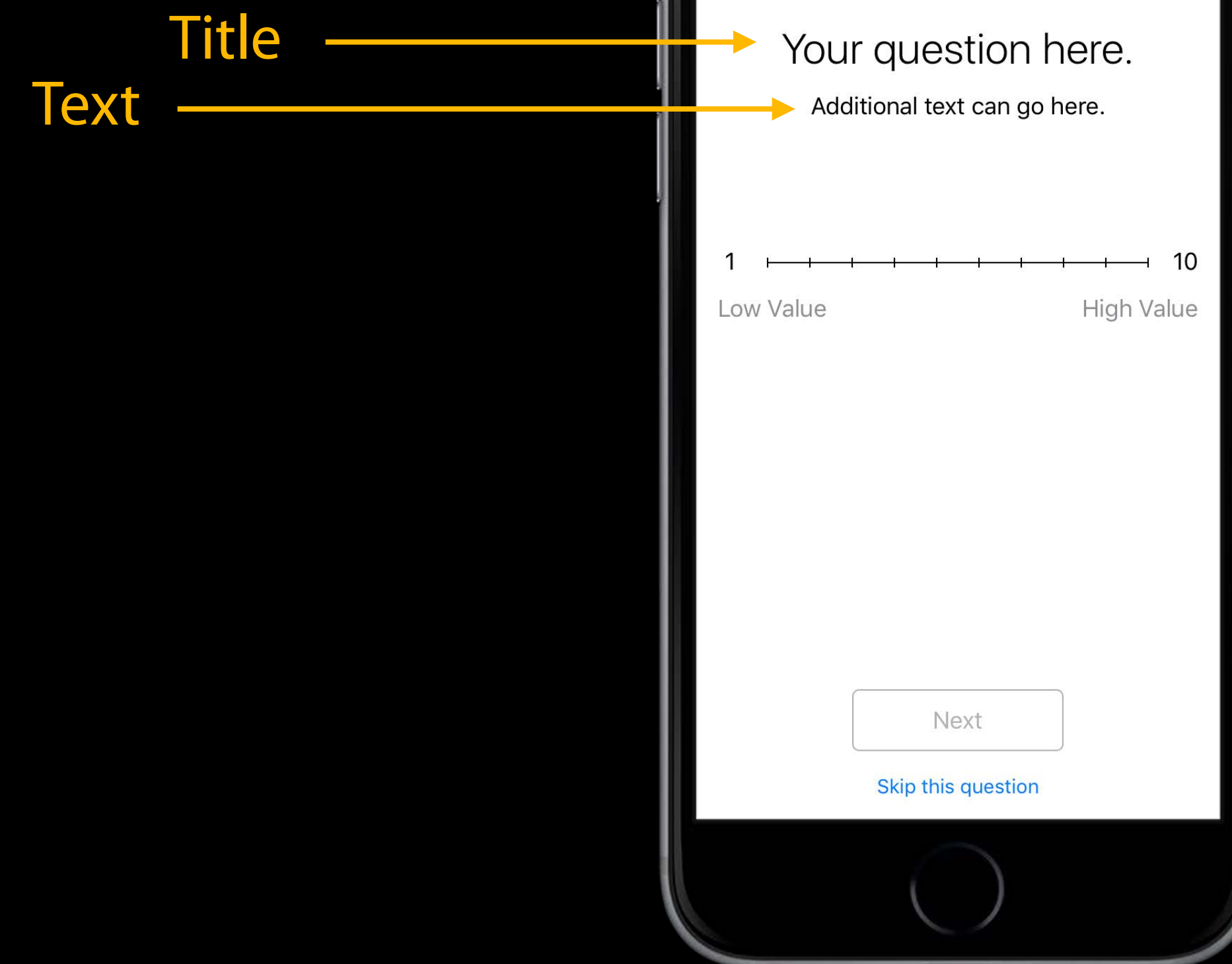
Title



# ORKStep

## Steps in code

```
class ORKStep : NSObject {  
  
    var identifier: String { get }  
  
    var title: String?  
    var text: String?
```

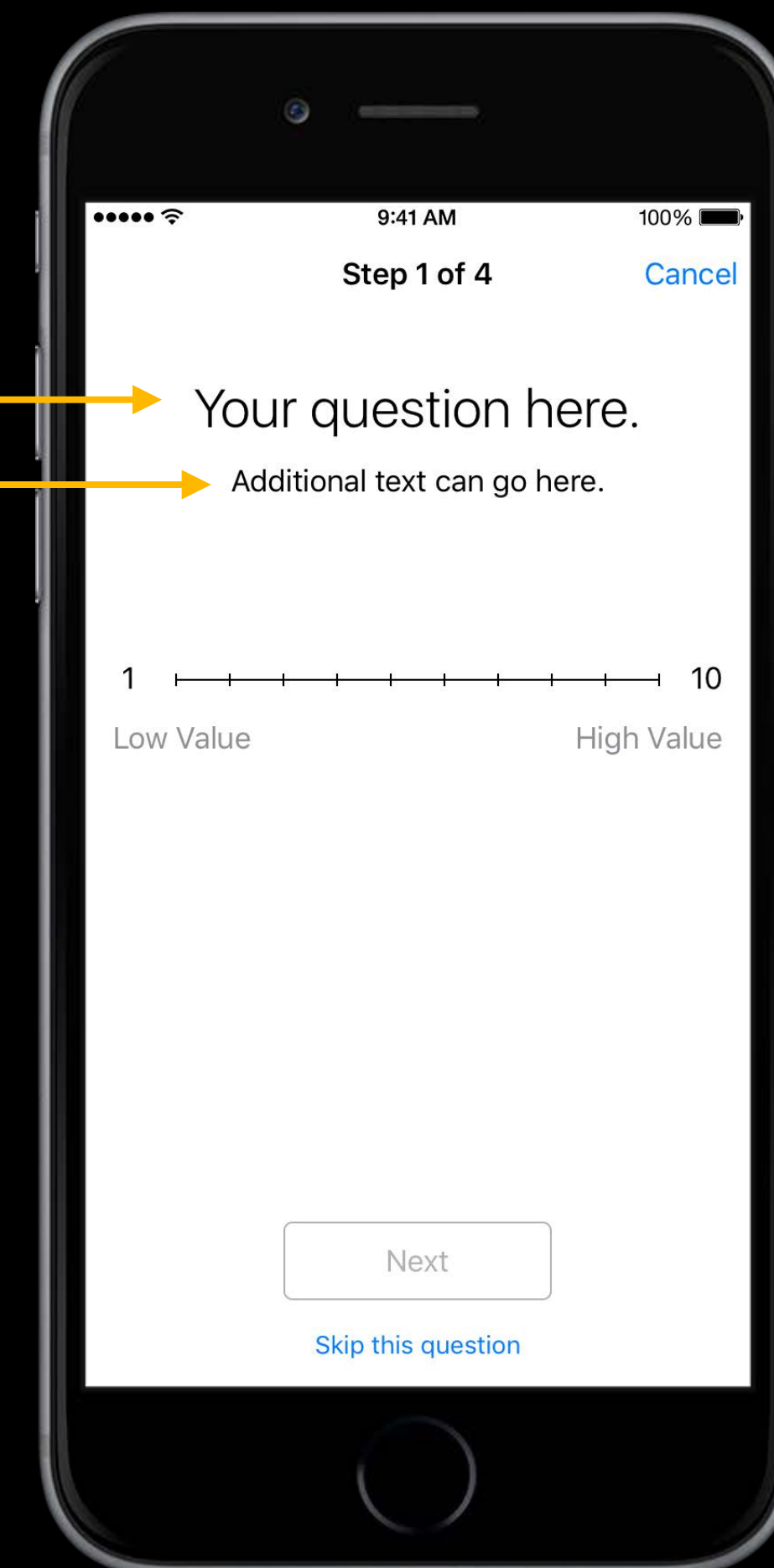


# ORKStep

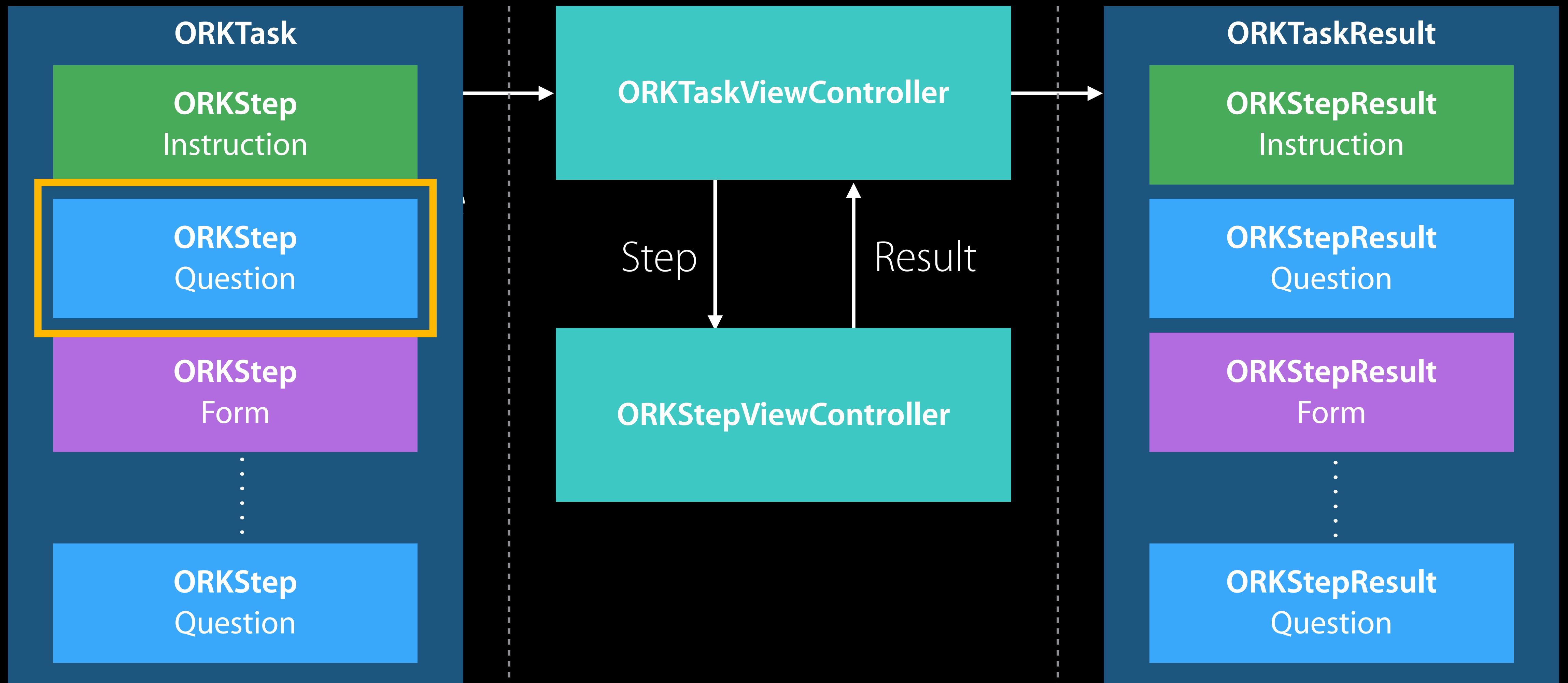
## Steps in code

```
class ORKStep : NSObject {  
  
    var identifier: String { get }  
  
    var title: String?  
    var text: String?  
  
    var optional: Bool  
    ...  
}
```

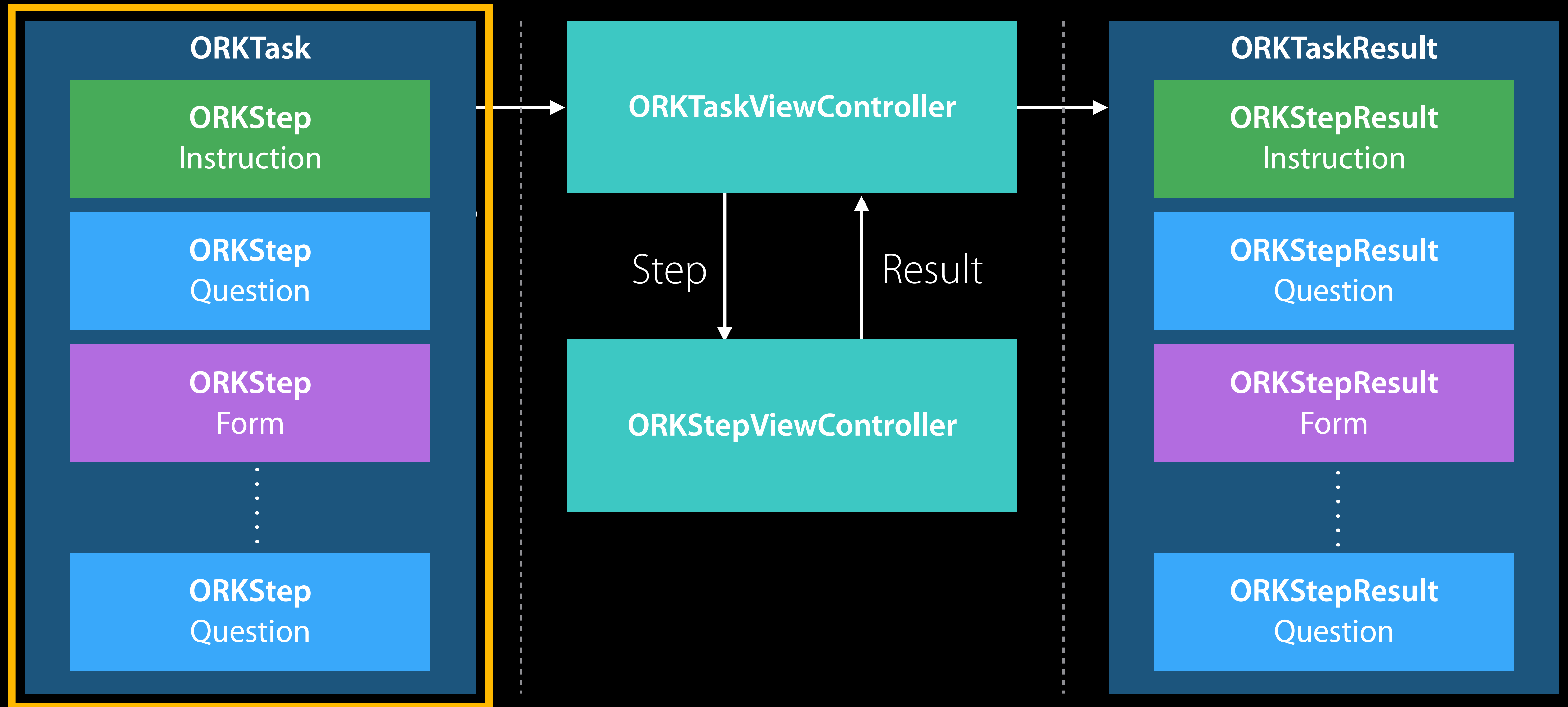
Title —————> Your question here.  
Text —————> Additional text can go here.



# Overview



# Overview



# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {
```

# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
}
```

# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
    func stepAfterStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?
```

# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
    func stepAfterStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    func stepBeforeStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    ...  
}
```

# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
    func stepAfterStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    func stepBeforeStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    ...  
}  
  
class ORKOrderedTask : ORKTask {
```

# ORKTask

Tasks link steps together

```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
    func stepAfterStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    func stepBeforeStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    ...  
}  
  
class ORKOrderedTask : ORKTask {  
    init(identifier: String, steps: [ORKStep])
```

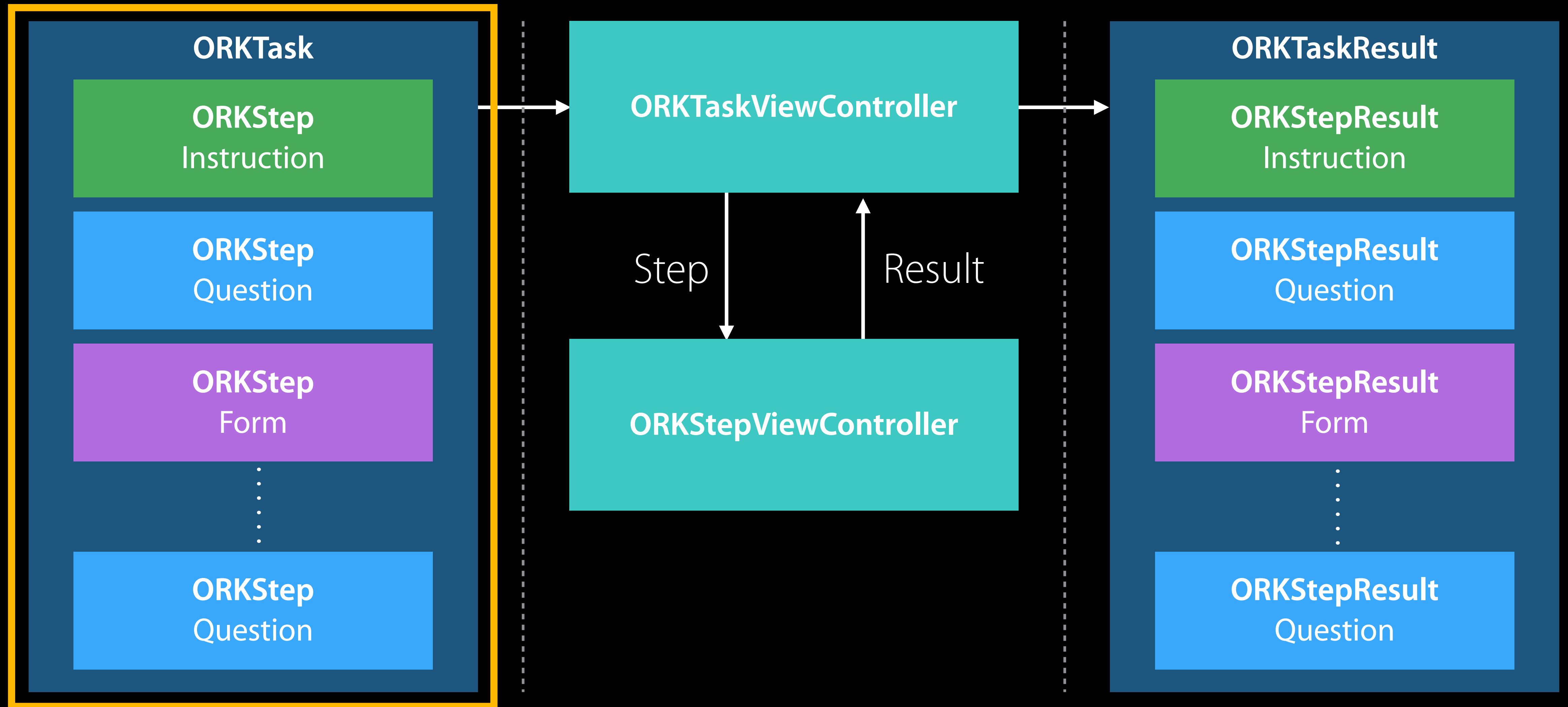
# ORKTask

Tasks link steps together

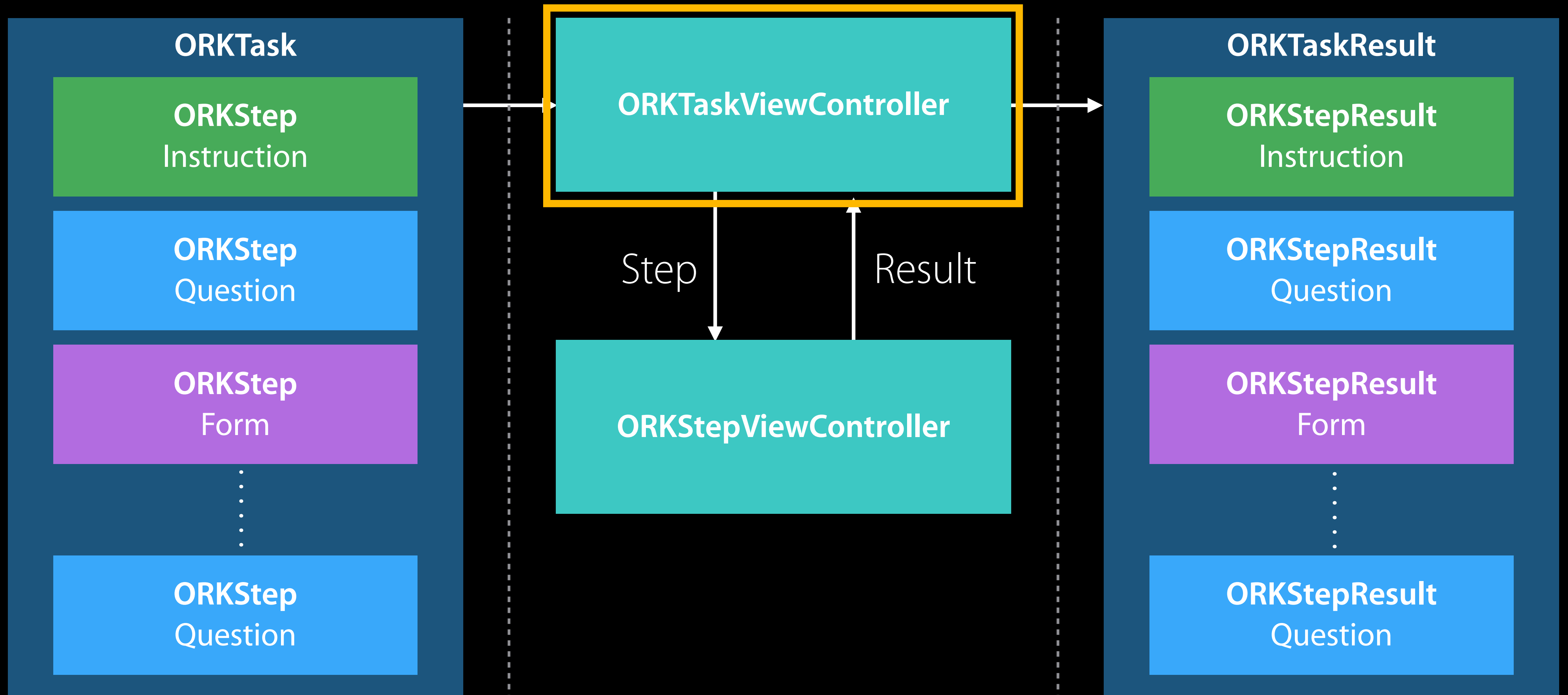
```
protocol ORKTask : NSObjectProtocol {  
    var identifier: String { get }  
    func stepAfterStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    func stepBeforeStep(step: ORKStep?,  
        withResult result: ORKTaskResult) -> ORKStep?  
    ...  
}
```

```
class ORKOrderedTask : ORKTask {  
    init(identifier: String, steps: [ORKStep])  
    var steps: [ORKStep] { get }  
    ...  
}
```

# Overview



# Overview



# ORKTaskViewController

For presenting tasks

```
let task = ...
```

# ORKTaskViewController

For presenting tasks

```
let task = ...  
let taskViewController =  
    ORKTaskViewController(task: task,  
        taskRunUUID: nil)
```

# ORKTaskViewController

For presenting tasks

```
let task = ...
```

```
let taskViewController =  
    ORKTaskViewController(task: task,  
        taskRunUUID: nil)
```

```
taskViewController.delegate = self
```

# ORKTaskViewController

For presenting tasks

```
let task = ...
```

```
let taskViewController =  
    ORKTaskViewController(task: task,  
        taskRunUUID: nil)
```

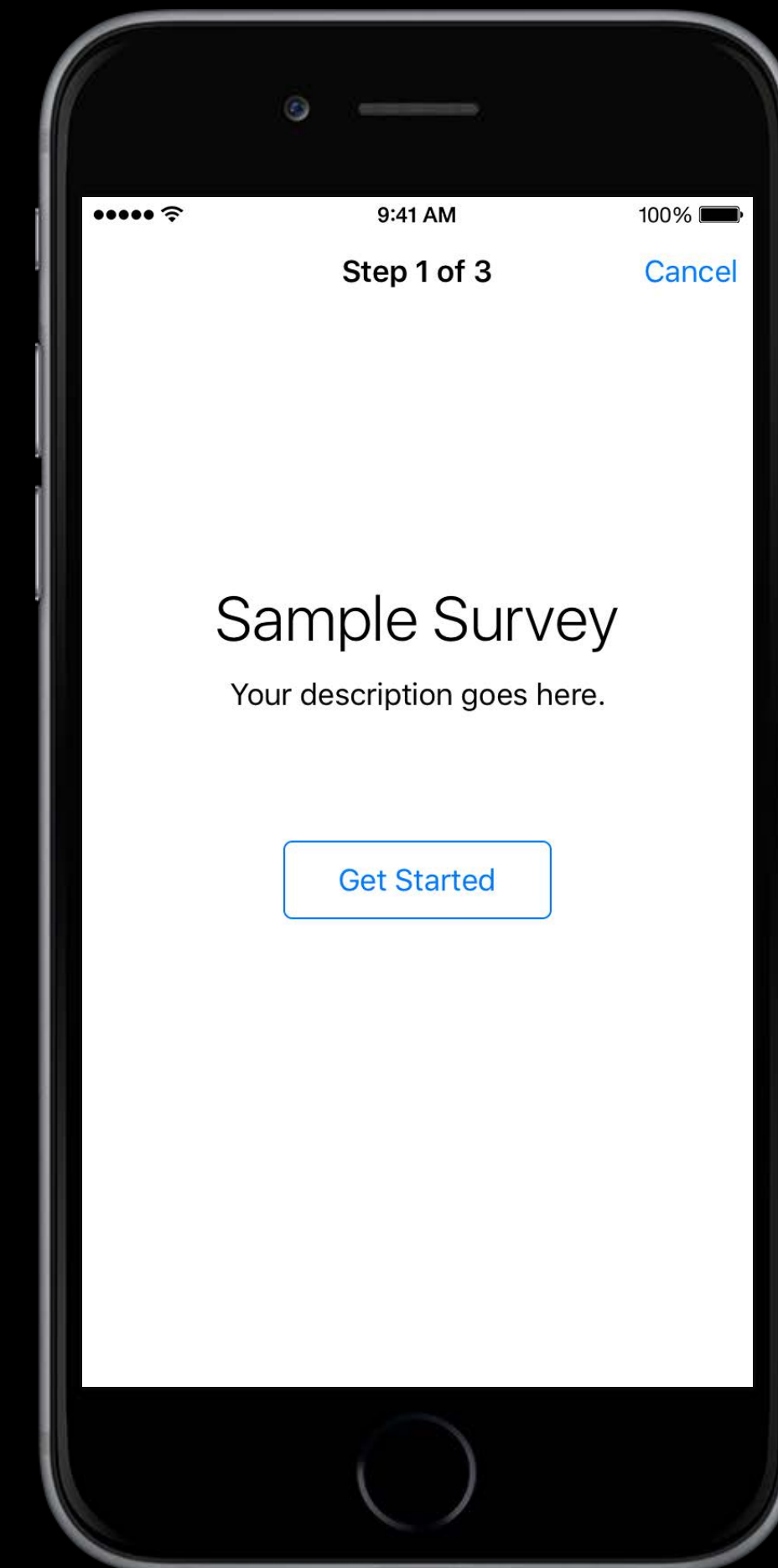
```
taskViewController.delegate = self
```

```
taskViewController.outputDirectory = ...
```

# ORKTaskViewController

For presenting tasks

```
let task = ...  
  
let taskViewController =  
    ORKTaskViewController(task: task,  
                           taskRunUUID: nil)  
  
taskViewController.delegate = self  
  
taskViewController.outputDirectory = ...  
  
presentViewController(taskViewController,  
                      animated: true,  
                      completion: nil)
```

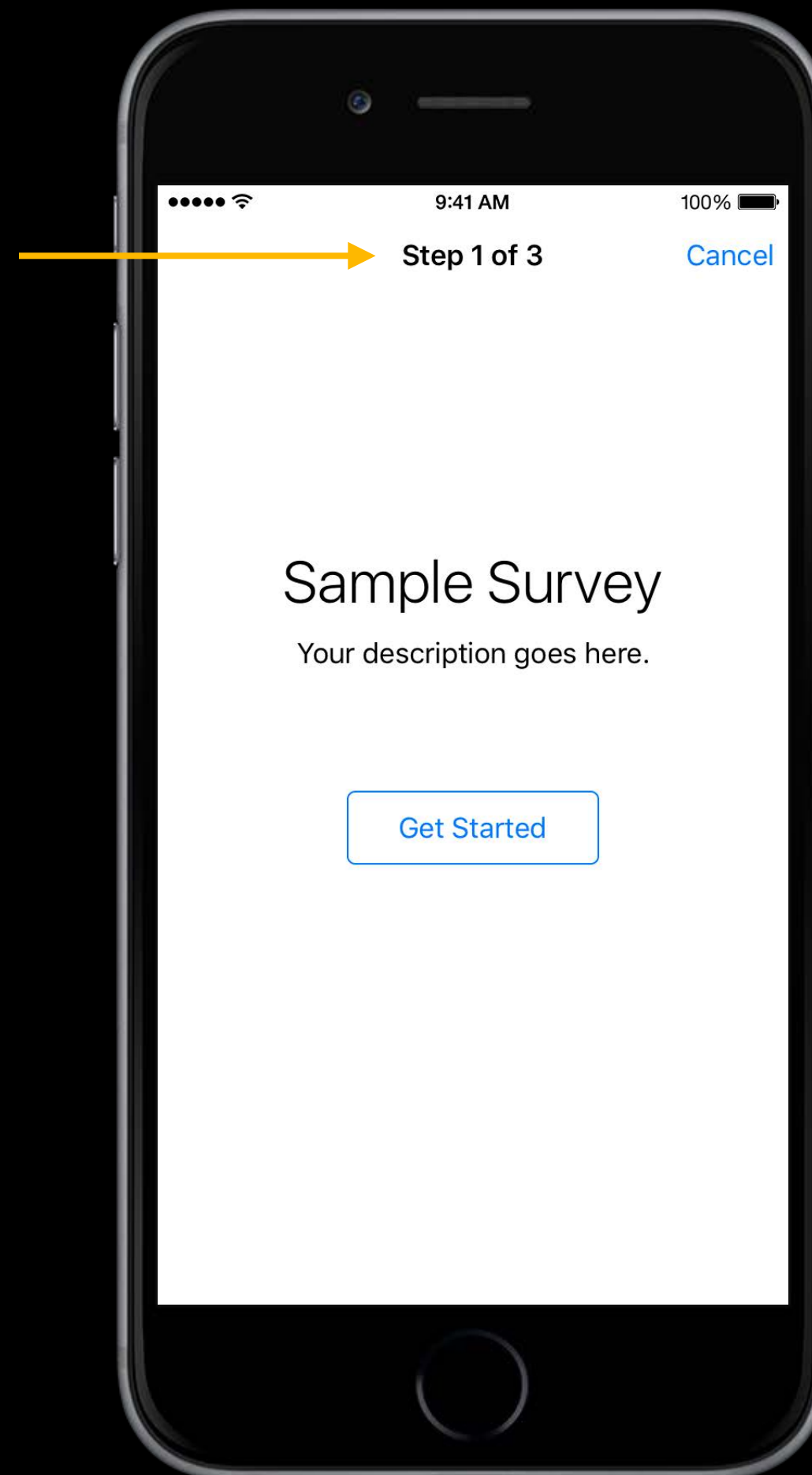


# ORKTaskViewController

For presenting tasks

```
let task = ...  
  
let taskViewController =  
    ORKTaskViewController(task: task,  
                           taskRunUUID: nil)  
  
taskViewController.delegate = self  
  
taskViewController.outputDirectory = ...  
  
presentViewController(taskViewController,  
                      animated: true,  
                      completion: nil)
```

Navigation  
and Progress



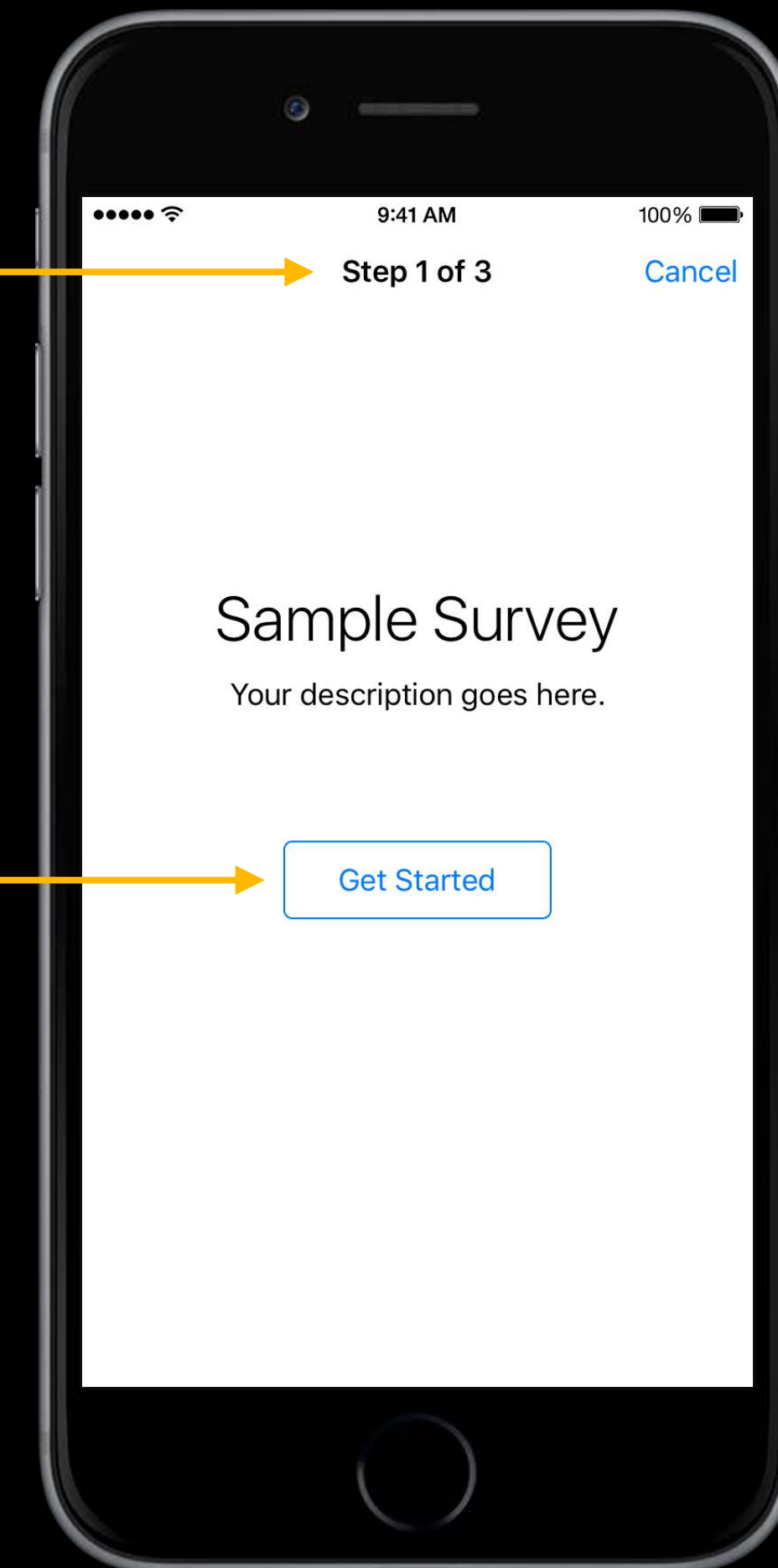
# ORKTaskViewController

For presenting tasks

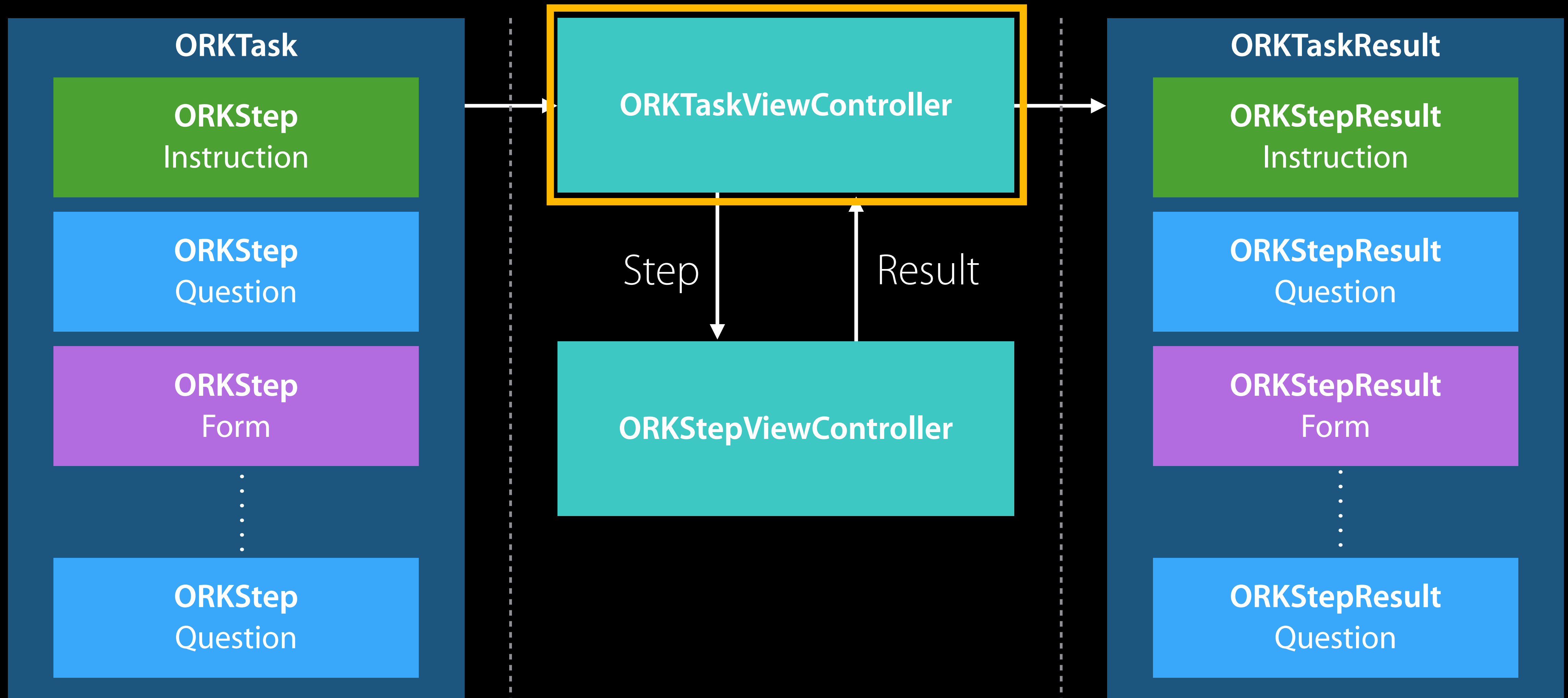
```
let task = ...  
  
let taskViewController =  
    ORKTaskViewController(task: task,  
                           taskRunUUID: nil)  
  
taskViewController.delegate = self  
  
taskViewController.outputDirectory = ...  
  
presentViewController(taskViewController,  
                      animated: true,  
                      completion: nil)
```

Navigation  
and Progress

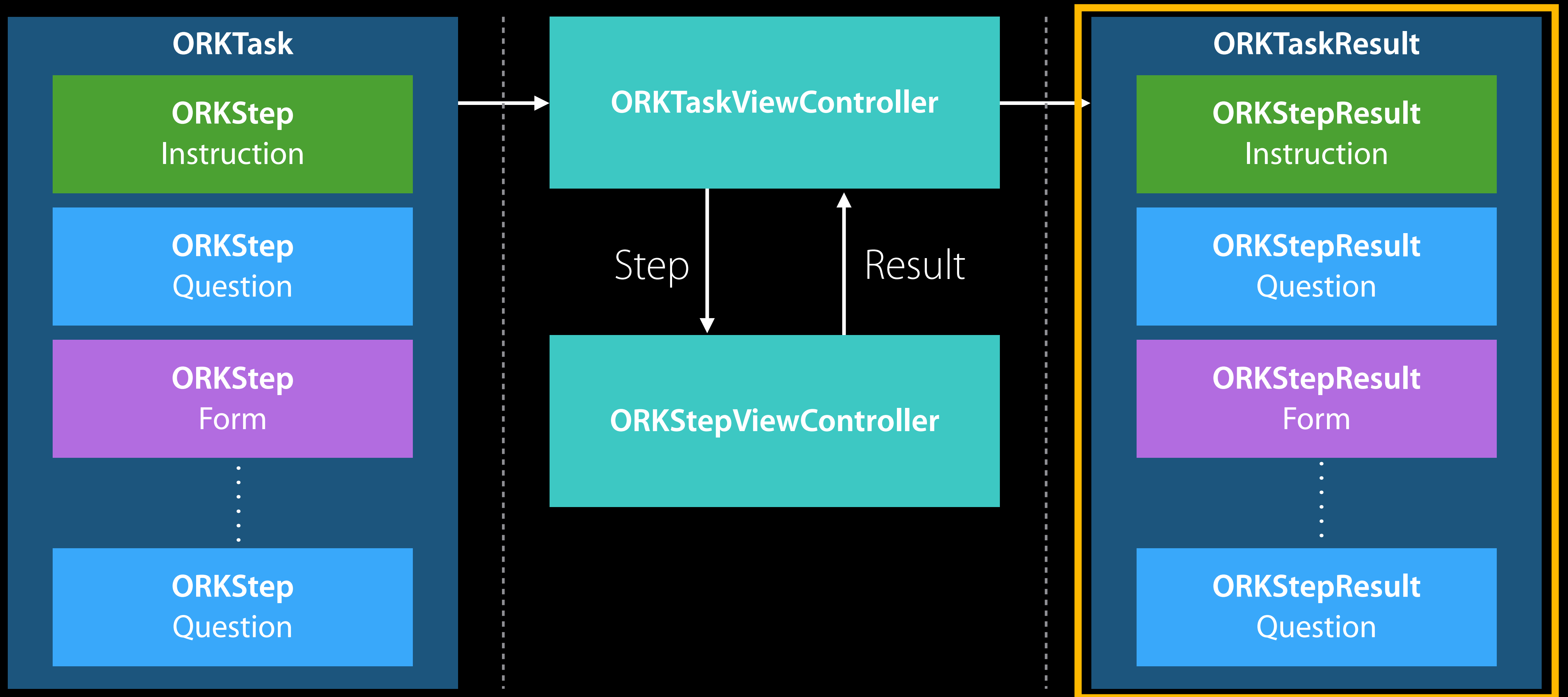
Forward  
Navigation



# Overview



# Overview



# Getting Results

When your task is done

```
func taskViewController(taskViewController: ORKTaskViewController,  
    didFinishWithReason reason: ORKTaskViewControllerFinishReason,  
    error: NSError?) {
```

# Getting Results

When your task is done

```
func taskViewController(taskViewController: ORKTaskViewController,
    didFinishWithReason reason: ORKTaskViewControllerFinishReason,
    error: NSError?) {
    if reason == .Saved {
        let restorationData = taskViewController.restorationData
        // Save it for later
        return
    }
}
```

# Getting Results

When your task is done

```
func taskViewController(taskViewController: ORKTaskViewController,
    didFinishWithReason reason: ORKTaskViewControllerFinishReason,
    error: NSError?) {
    if reason == .Saved {
        let restorationData = taskViewController.restorationData
        // Save it for later
        return
    }

    let taskResult = taskViewController.result
    // Do stuff with the taskResult
```

# Getting Results

When your task is done

```
func taskViewController(taskViewController: ORKTaskViewController,
    didFinishWithReason reason: ORKTaskViewControllerFinishReason,
    error: NSError?) {
    if reason == .Saved {
        let restorationData = taskViewController.restorationData
        // Save it for later
        return
    }

    let taskResult = taskViewController.result
    // Do stuff with the taskResult

    taskViewController.dismissViewControllerAnimated(true, completion: nil)
}
```

# Surveys

Instructions

Questions

Forms

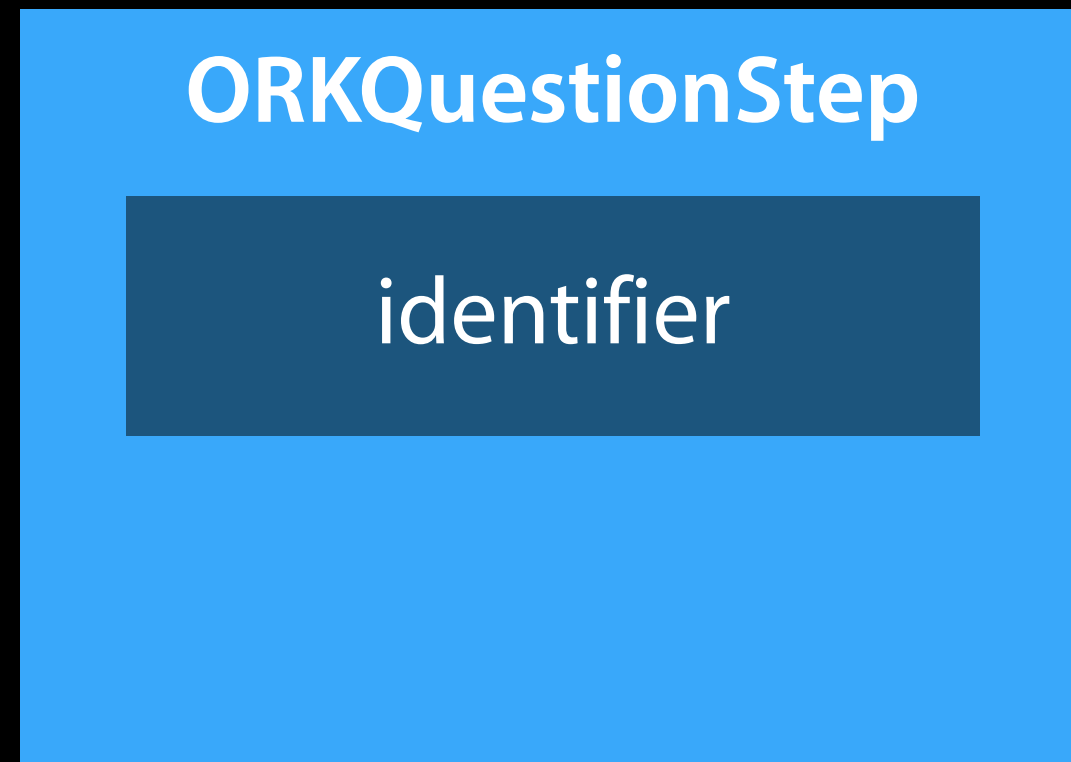
# Survey Steps

ORKInstructionStep

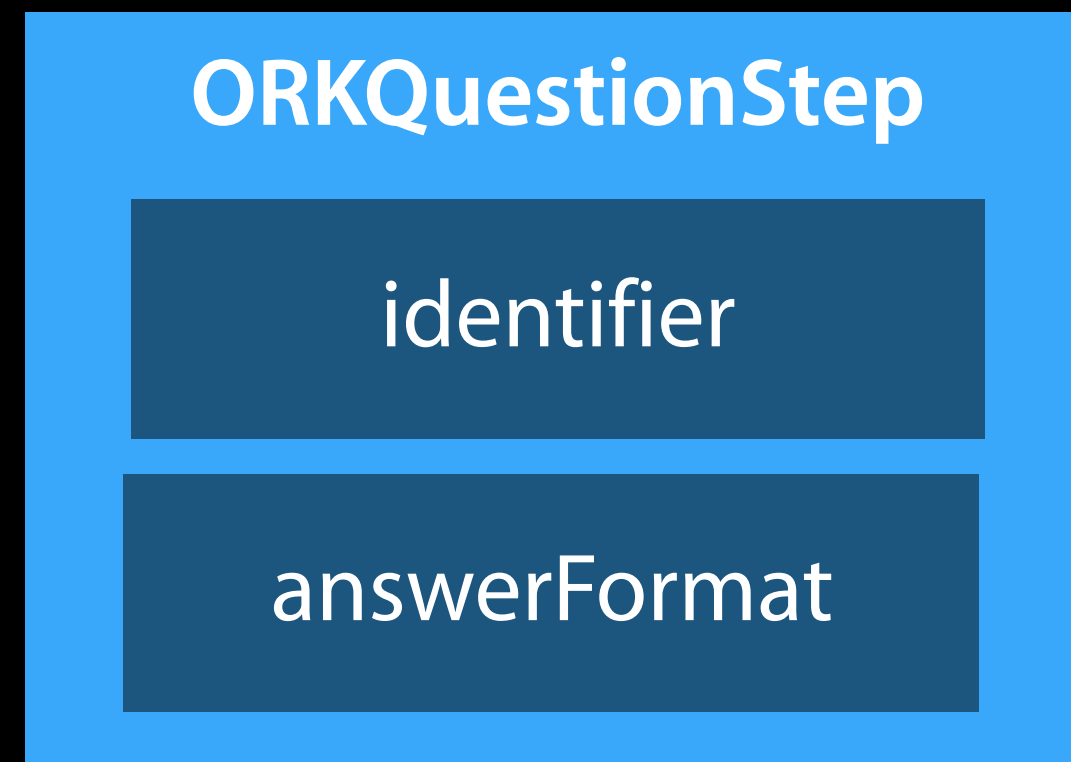
ORKQuestionStep

ORKFormStep

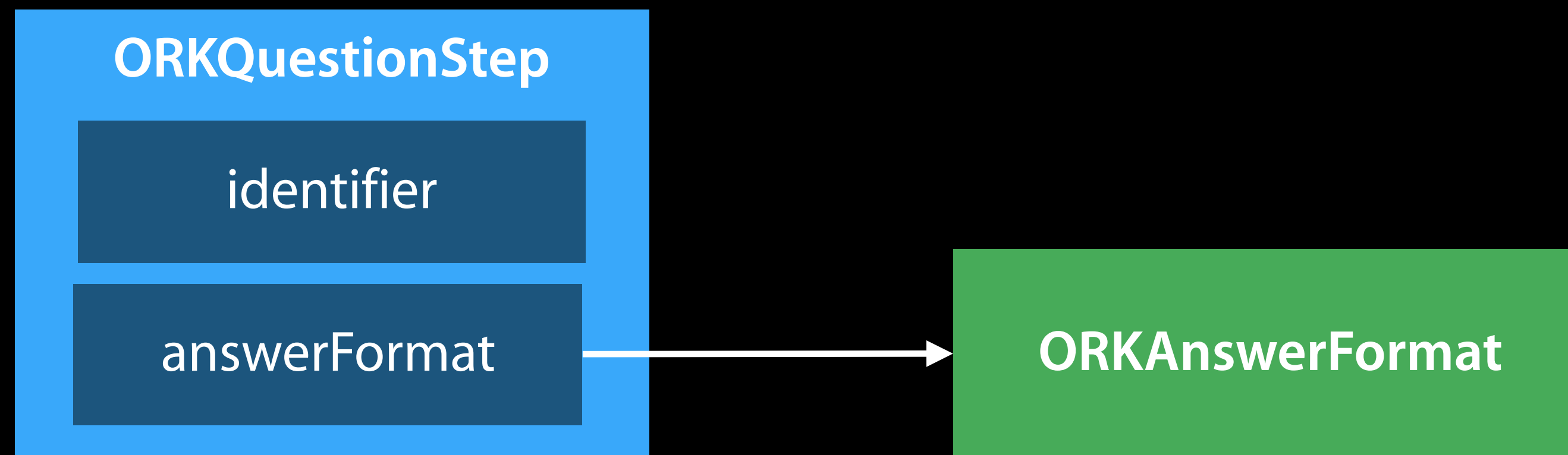
# Question Steps



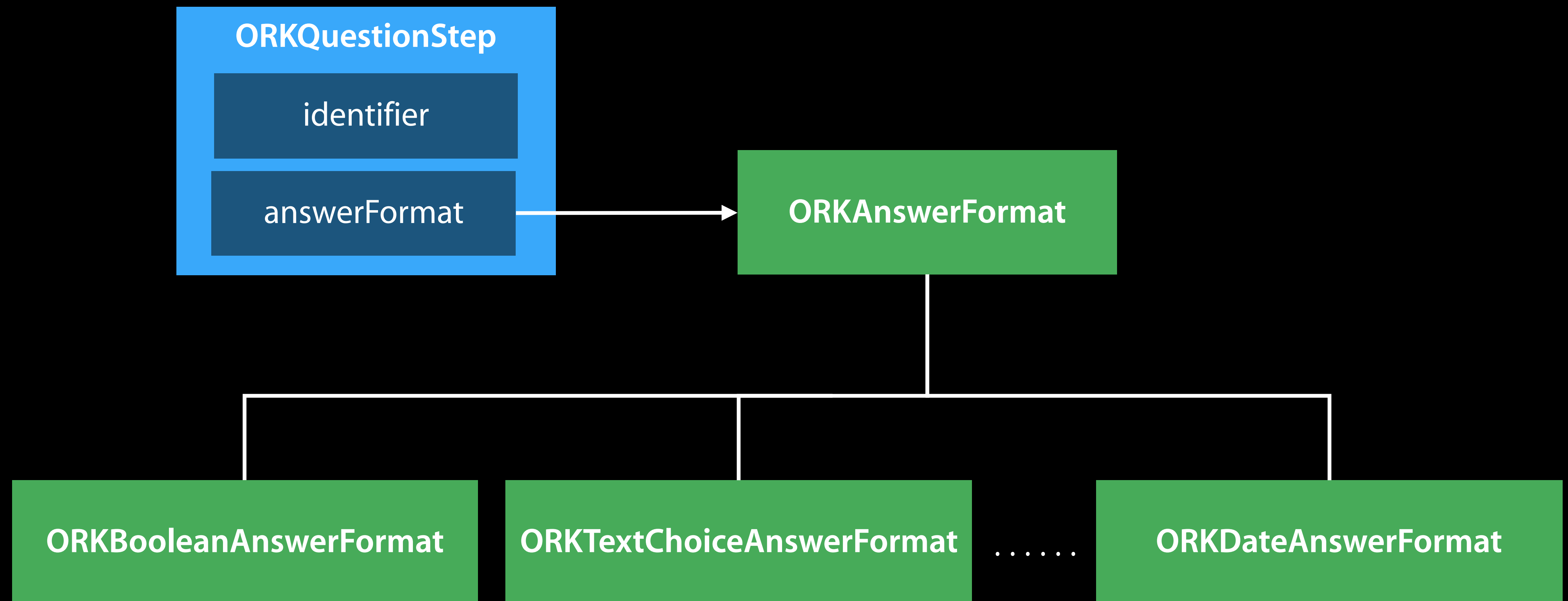
# Question Steps



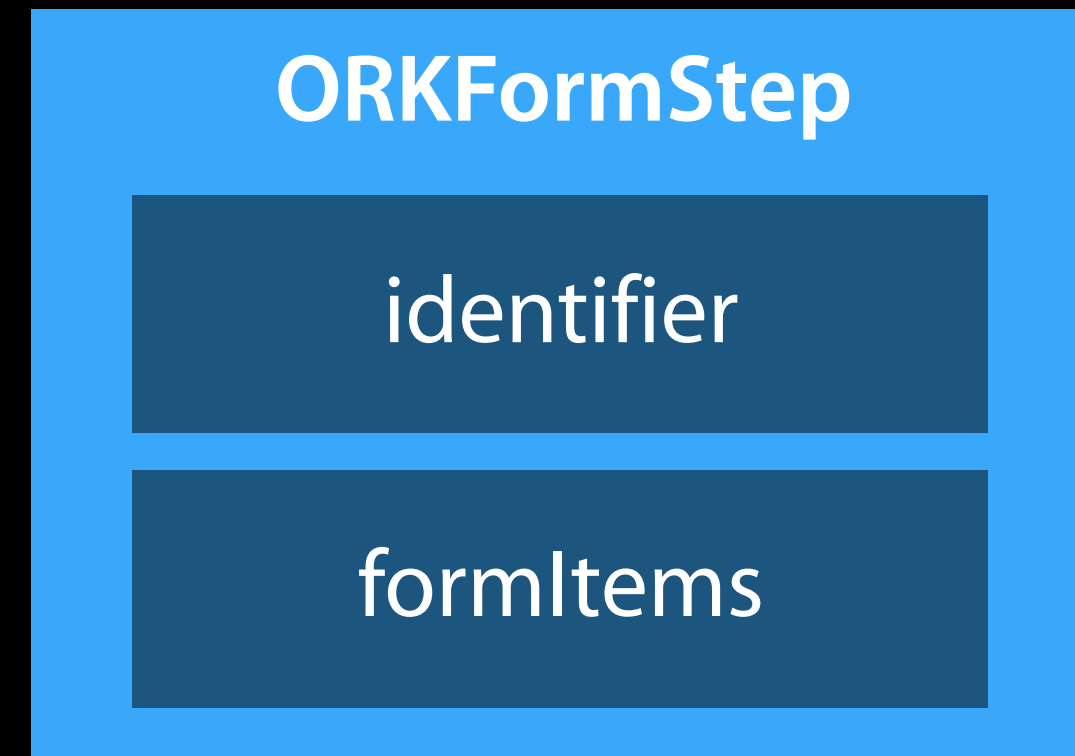
# Question Steps



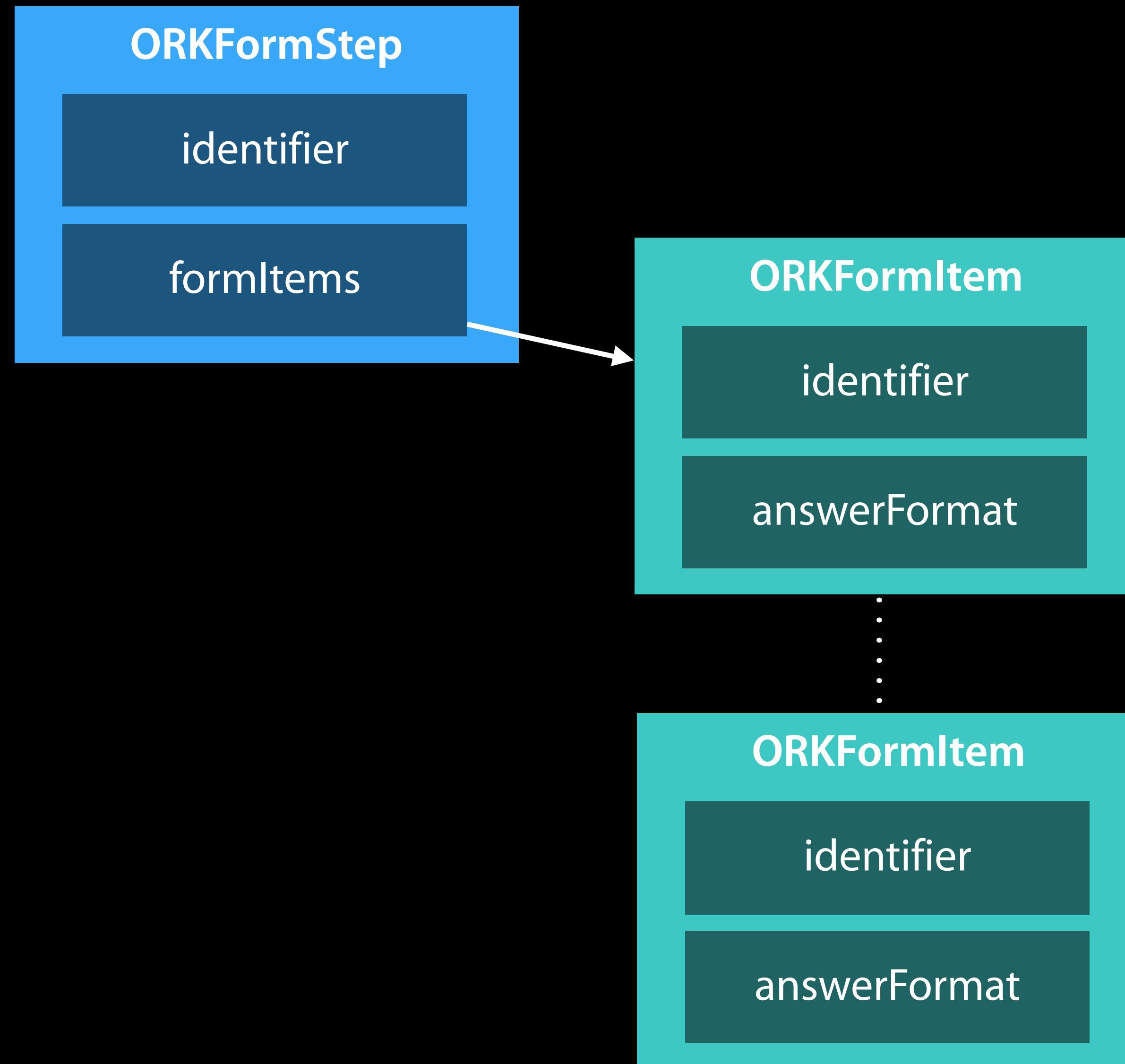
# Question Steps



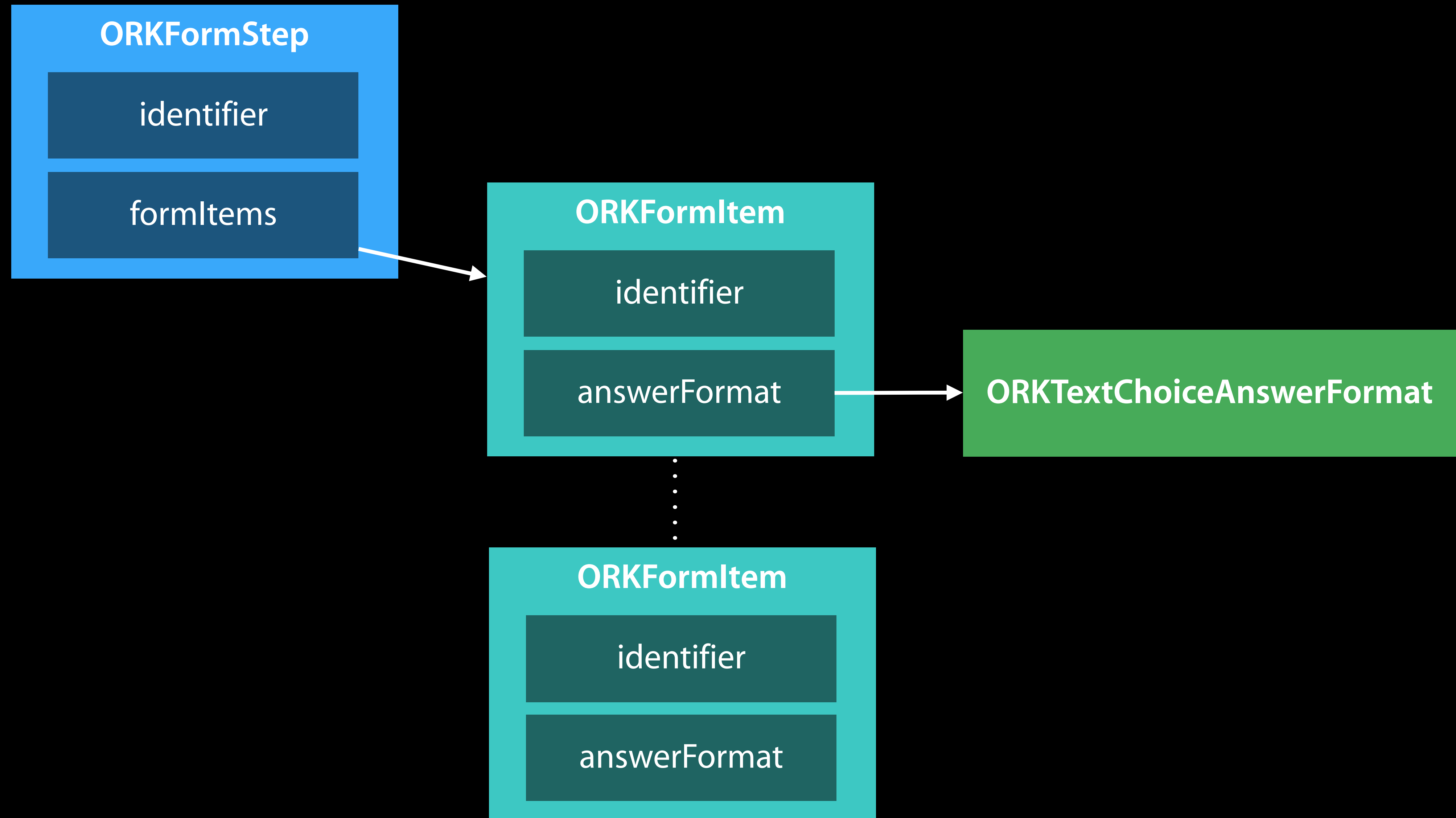
# Forms and Form Items



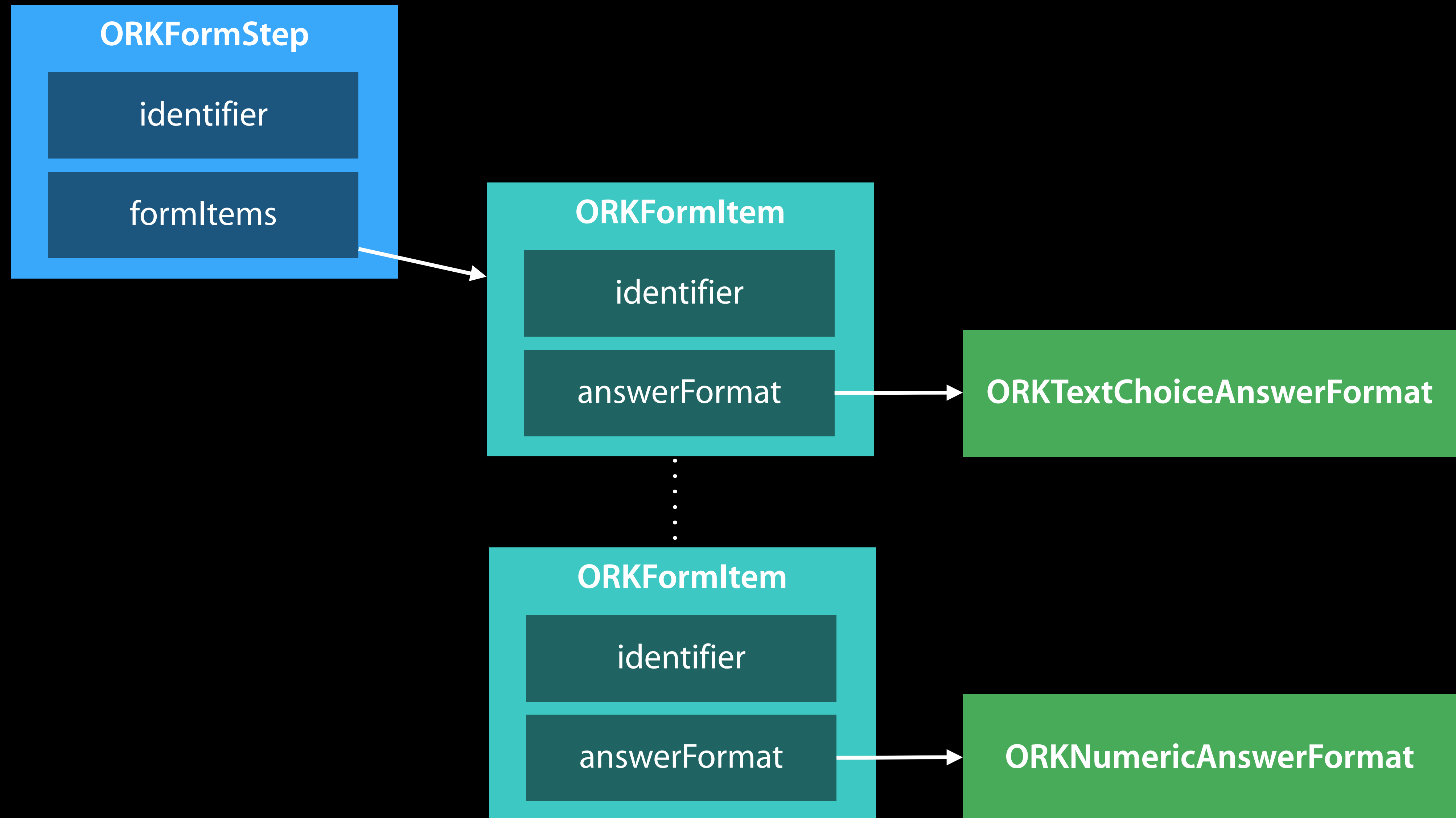
# Forms and Form Items



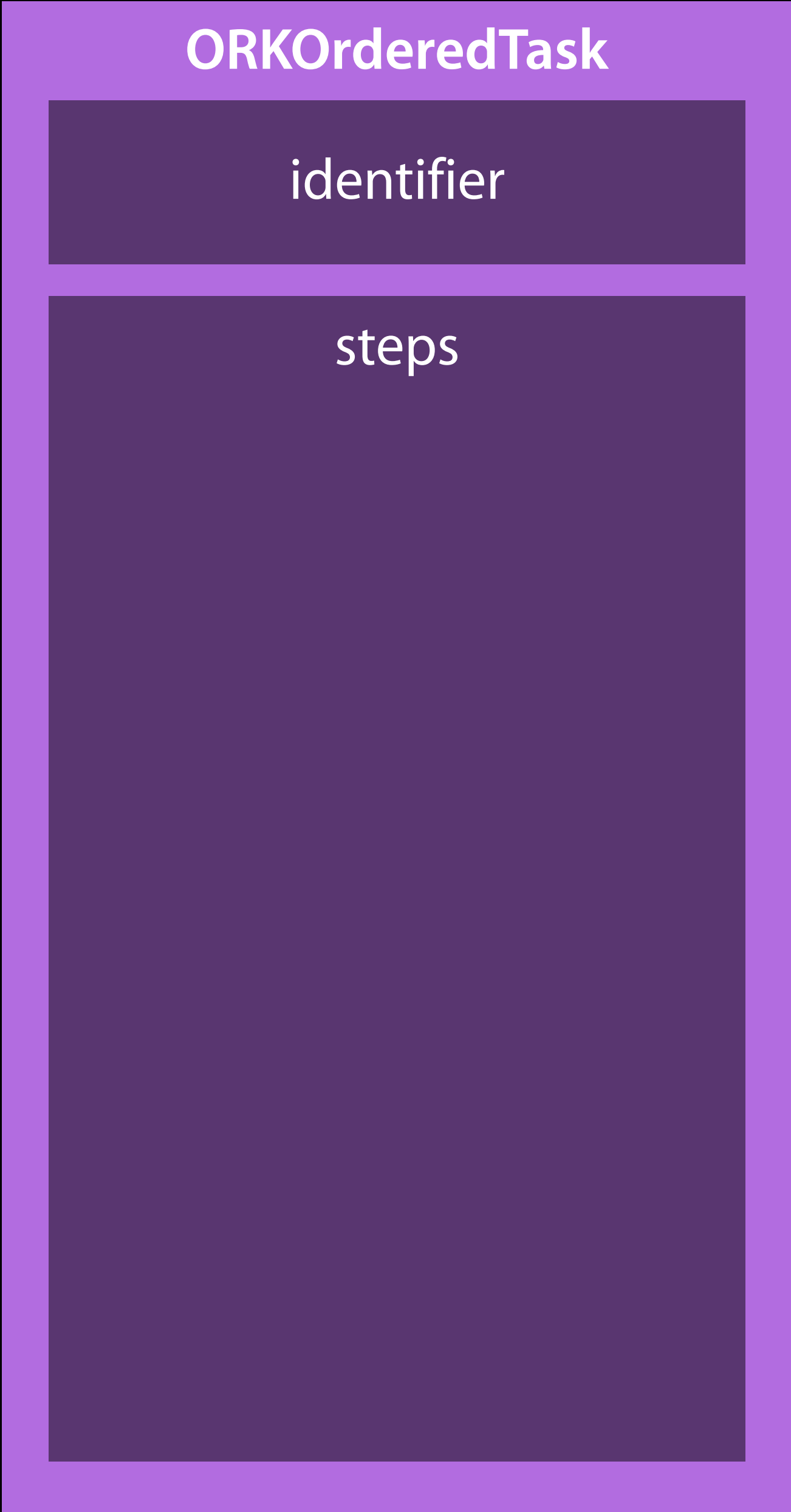
# Forms and Form Items



# Forms and Form Items



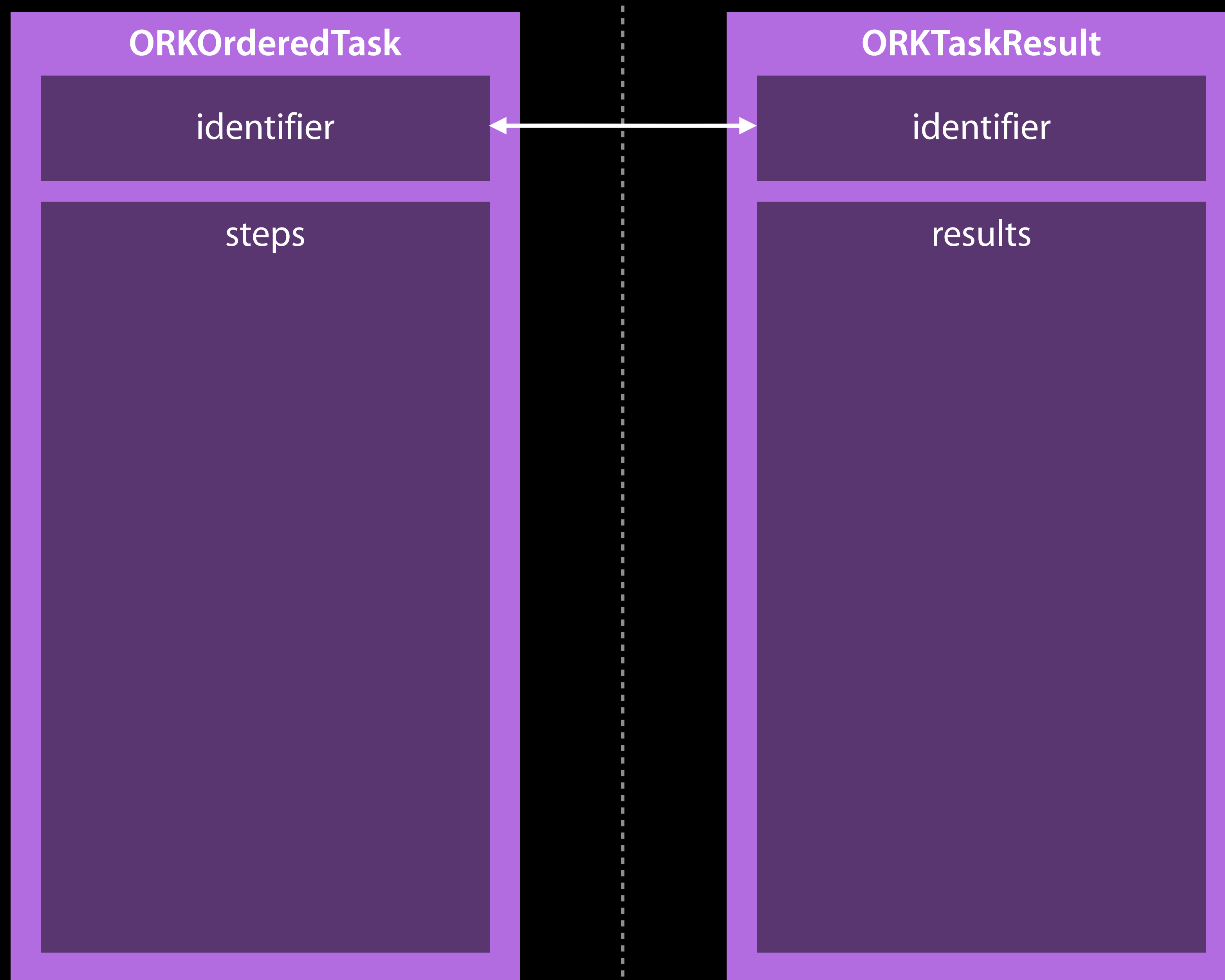
# Question Results



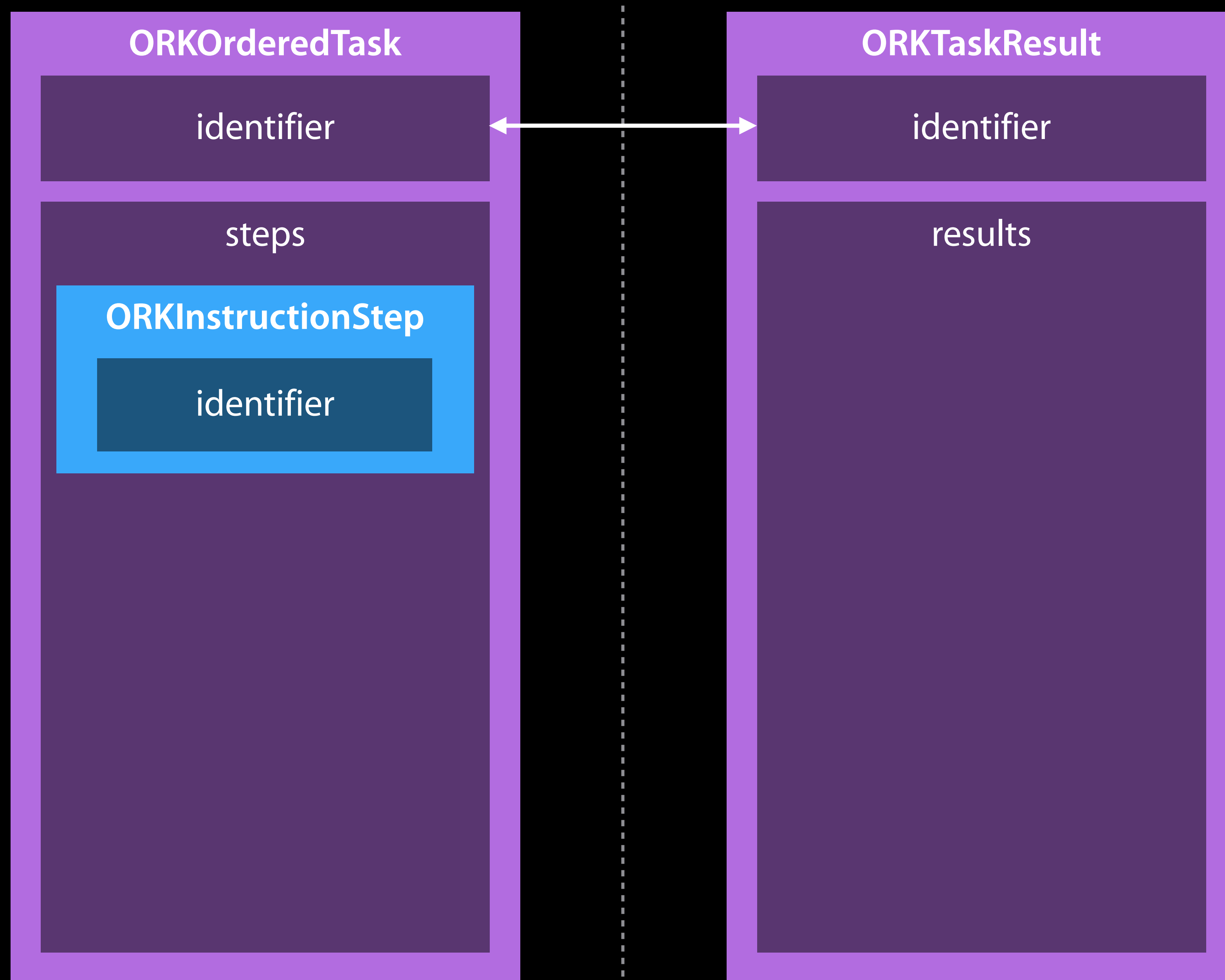
# Question Results



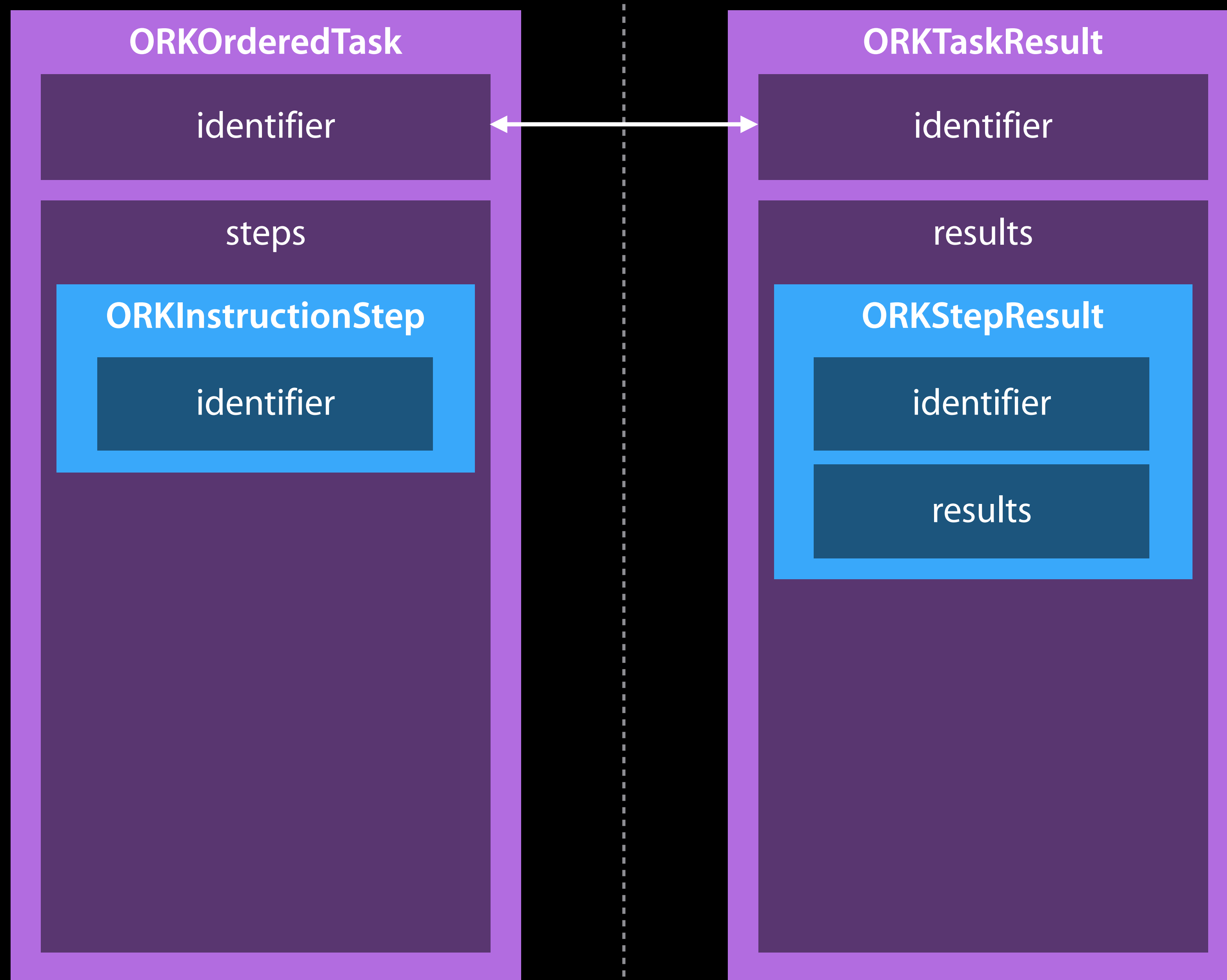
# Question Results



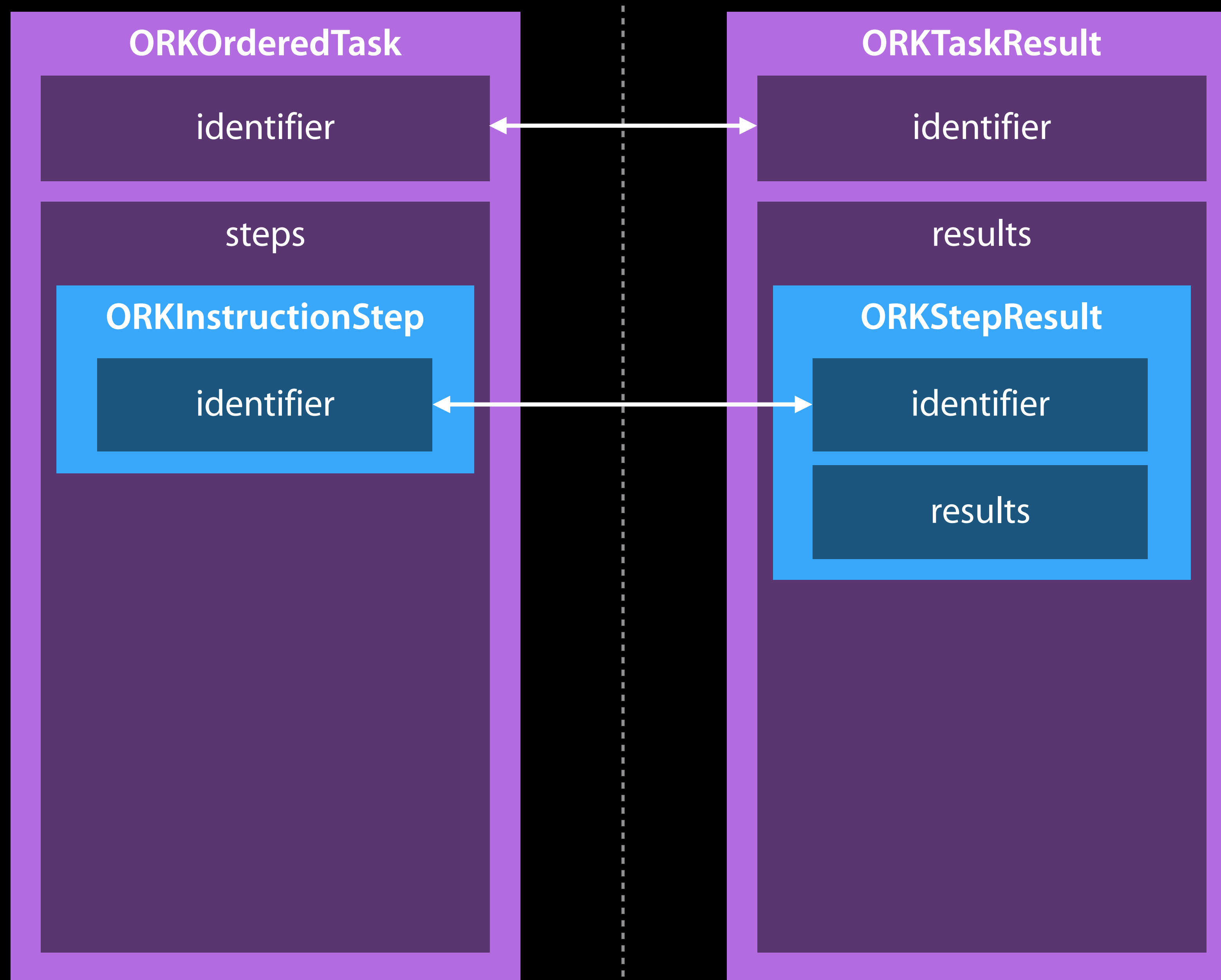
# Question Results



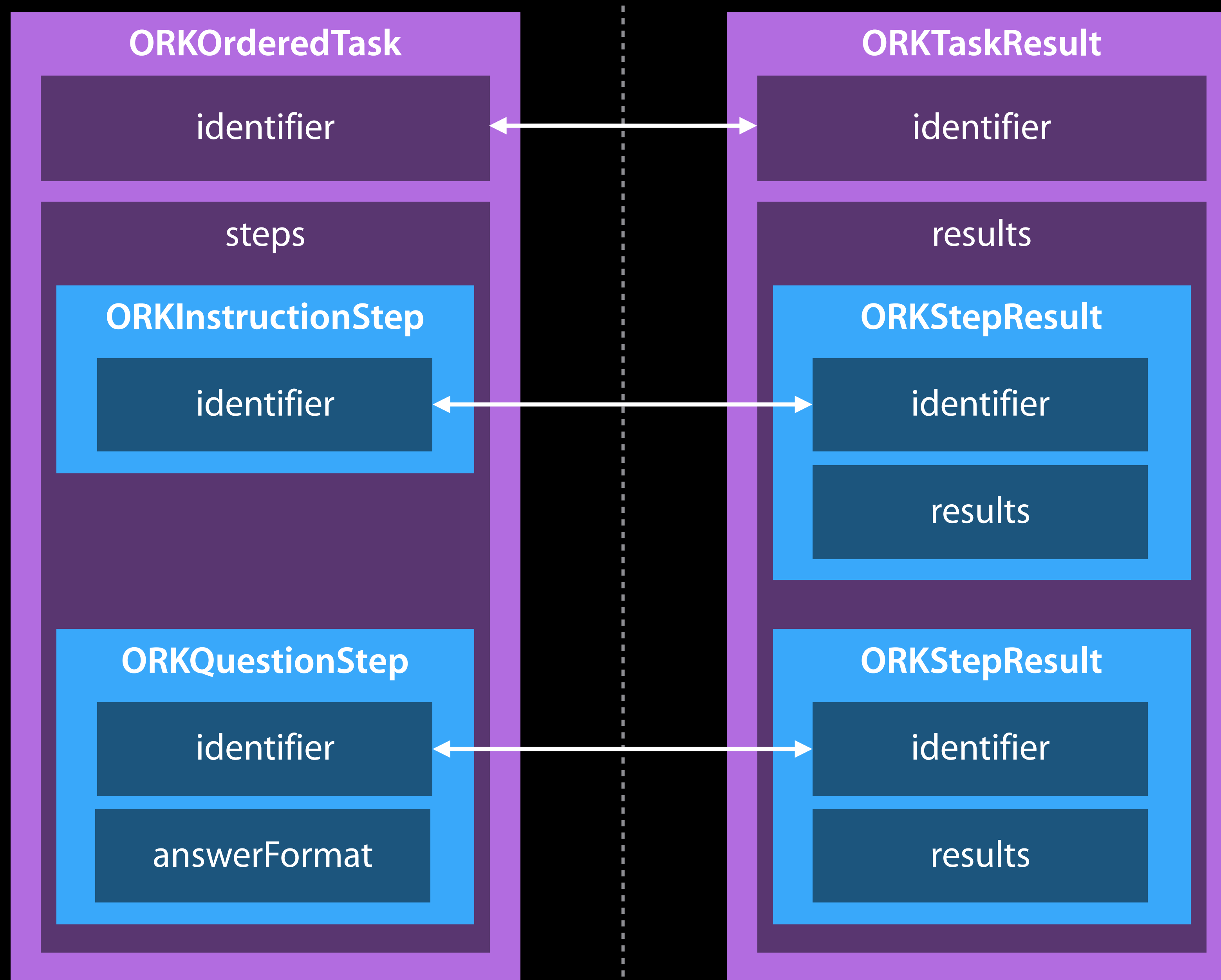
# Question Results



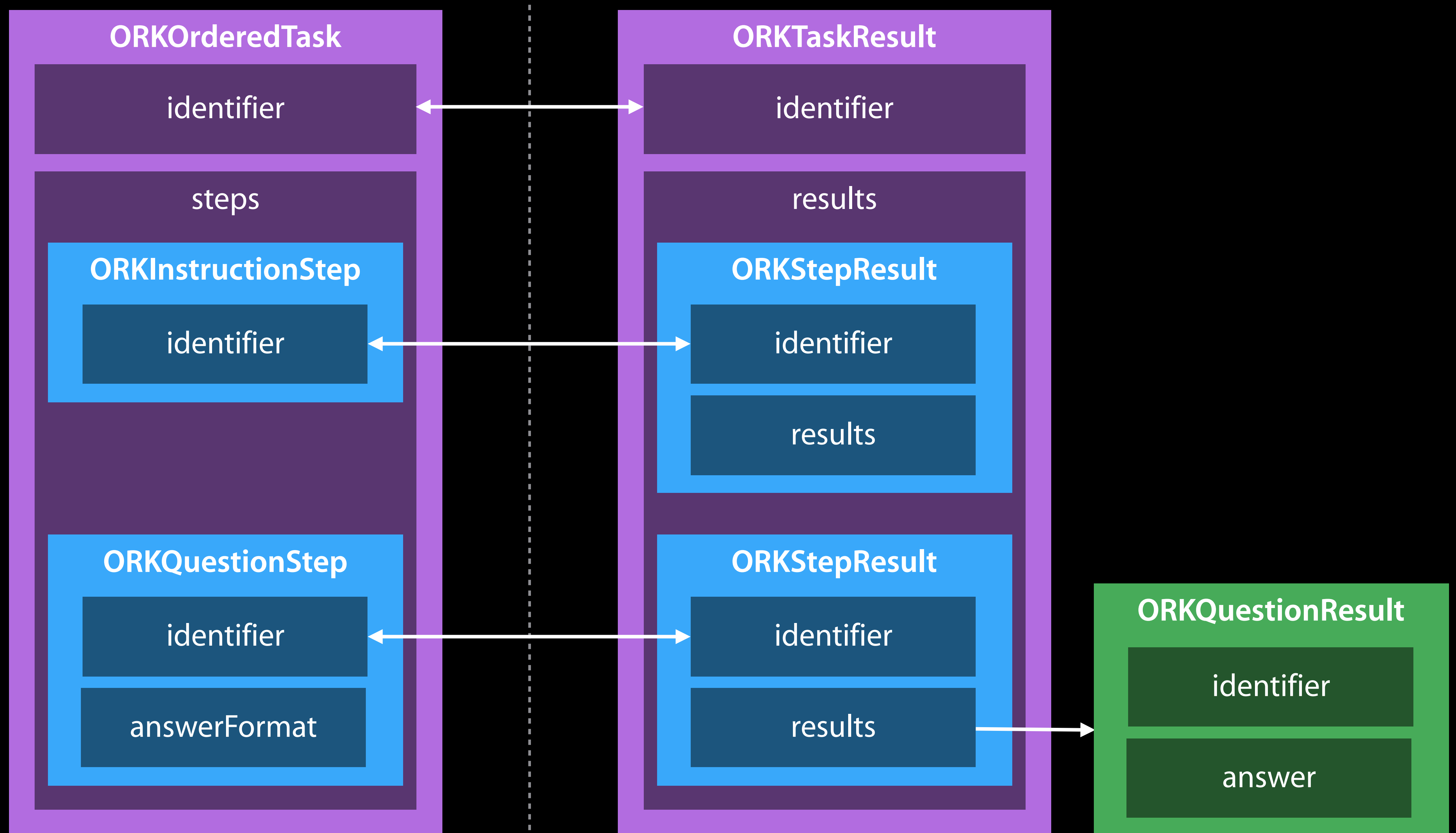
# Question Results



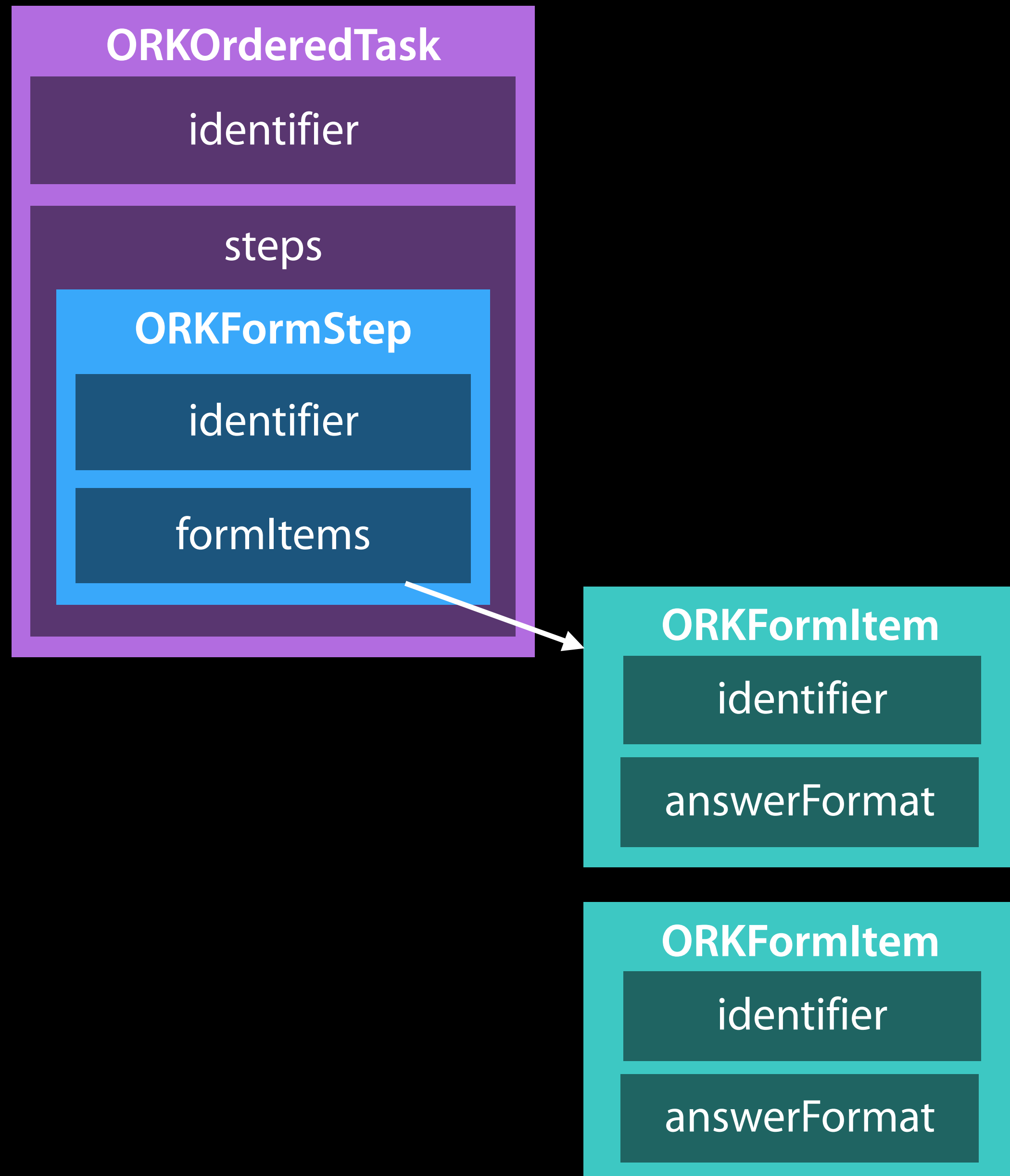
# Question Results



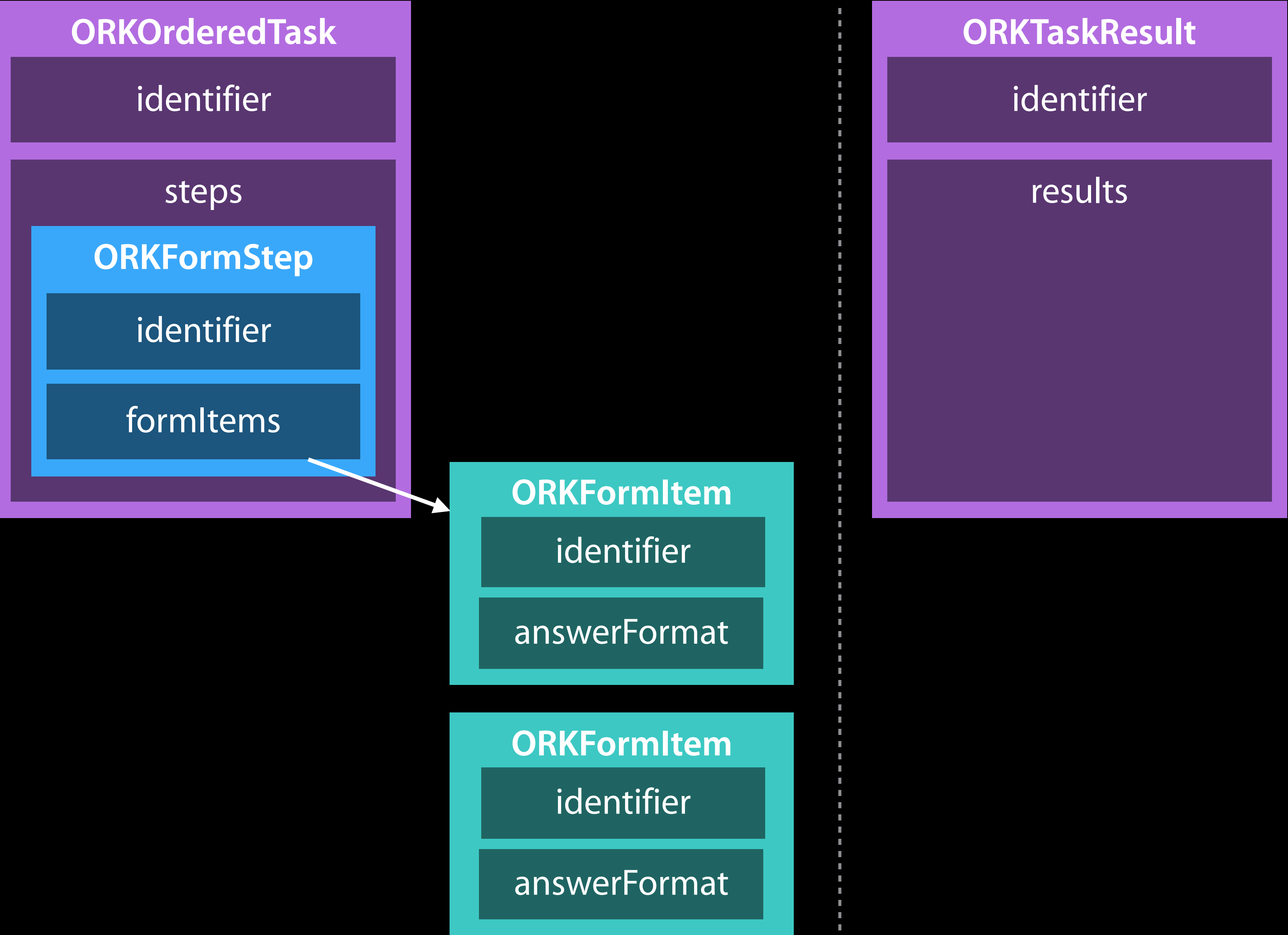
# Question Results



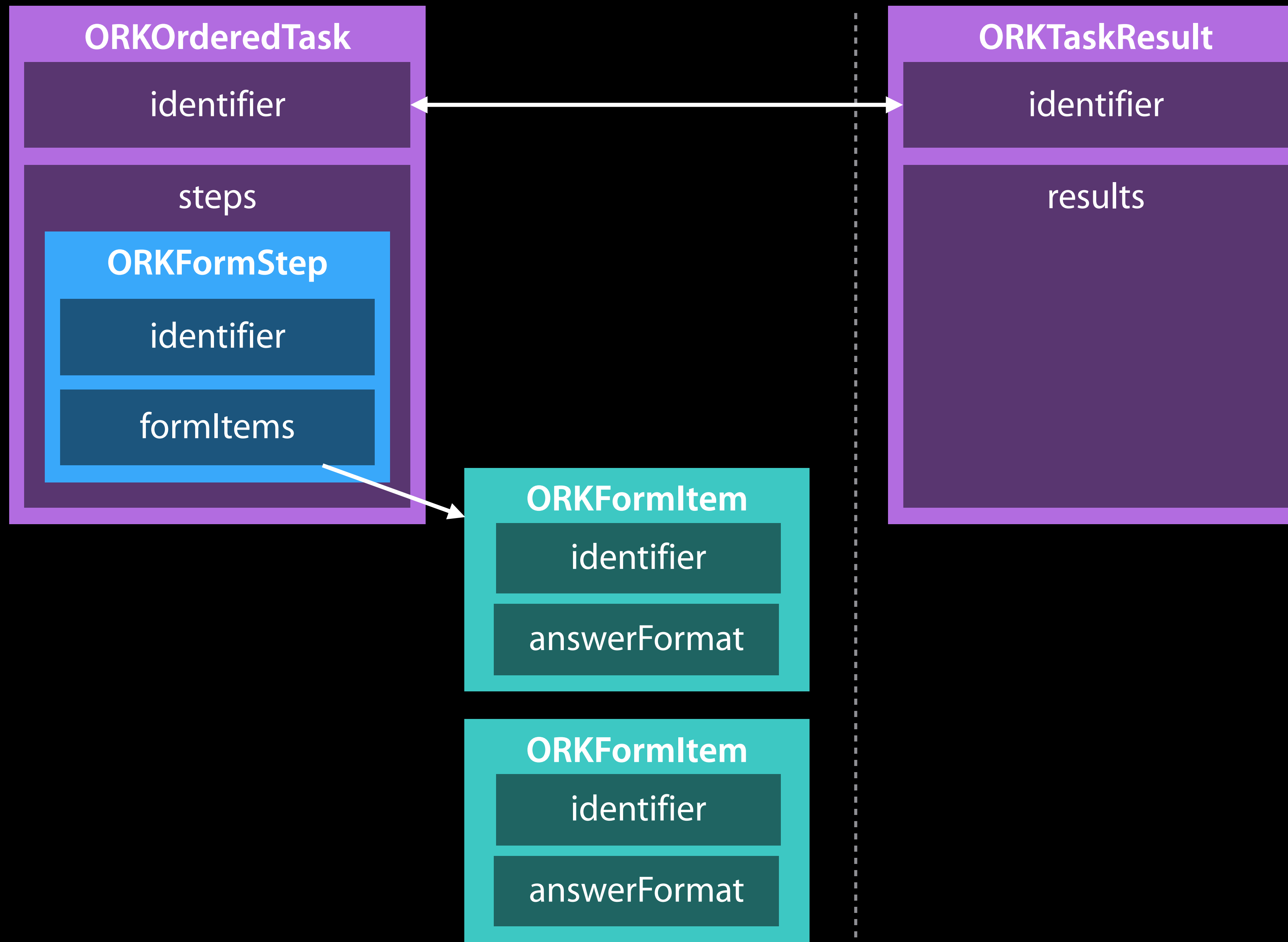
# Form Results



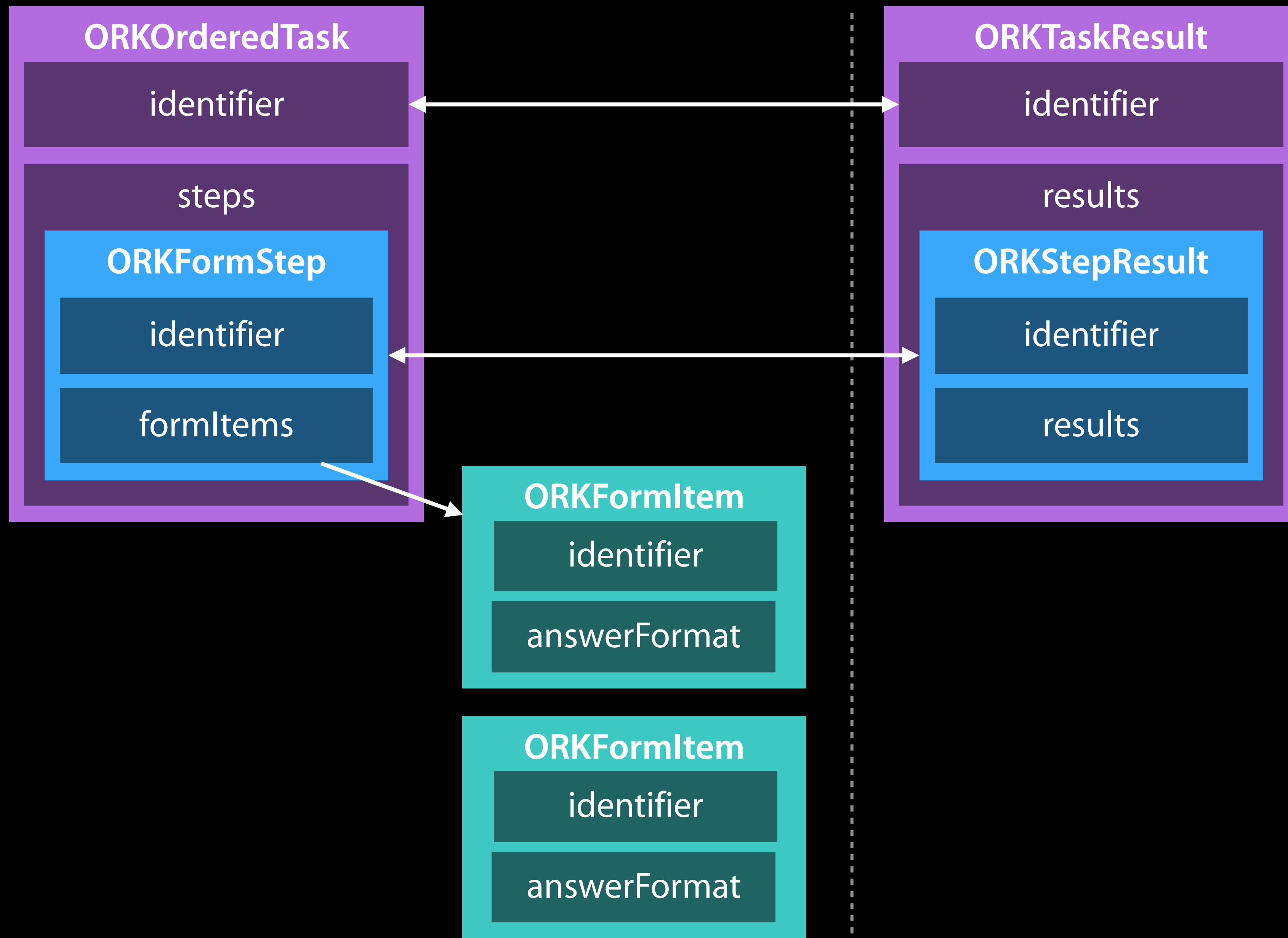
# Form Results



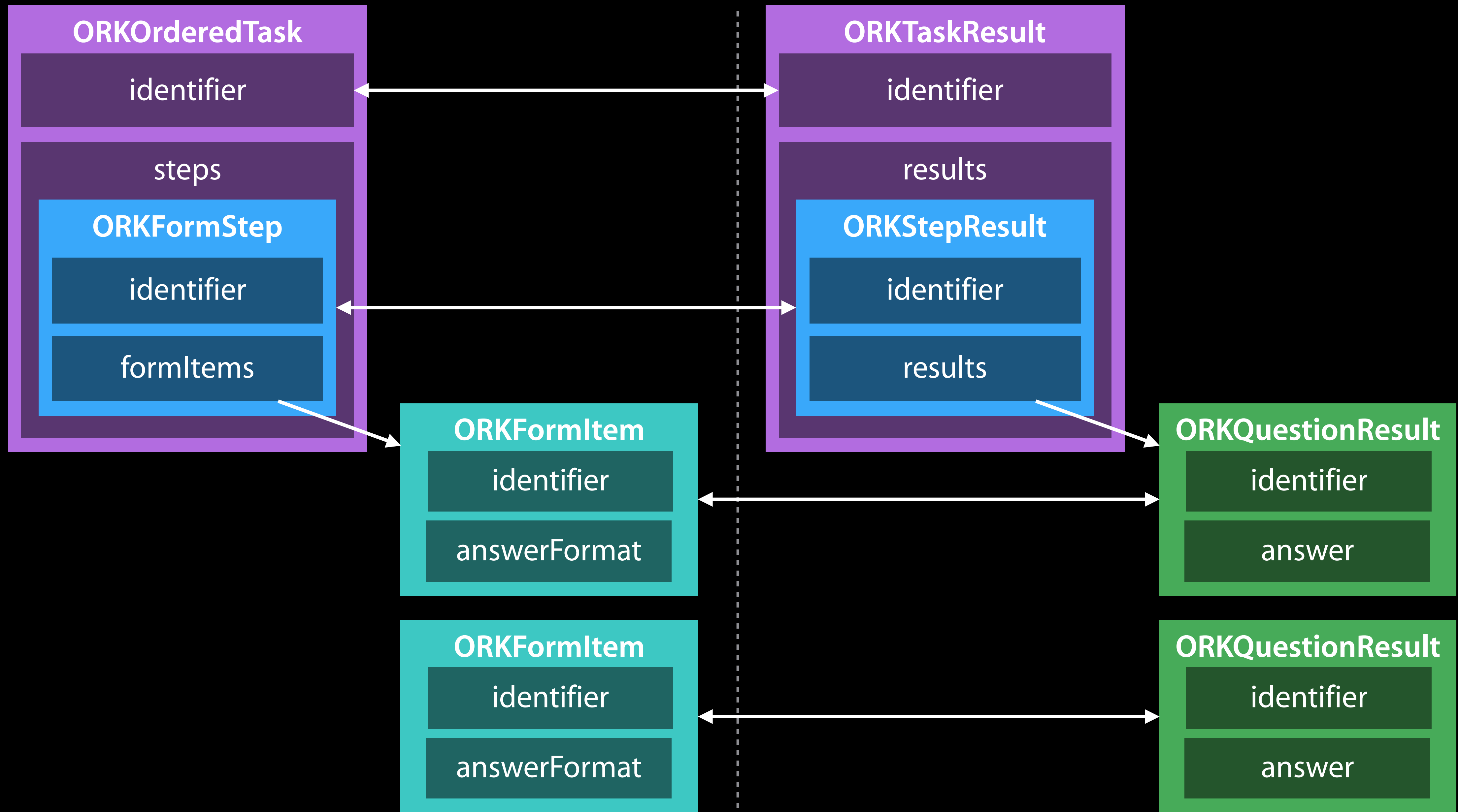
# Form Results



# Form Results



# Form Results



*Demo*

Surveys

# Informed Consent

What is it?

# What is it?

# What study entails

Page 10 of 10

# What is it?

## Risks and benefits

Page 10 of 10

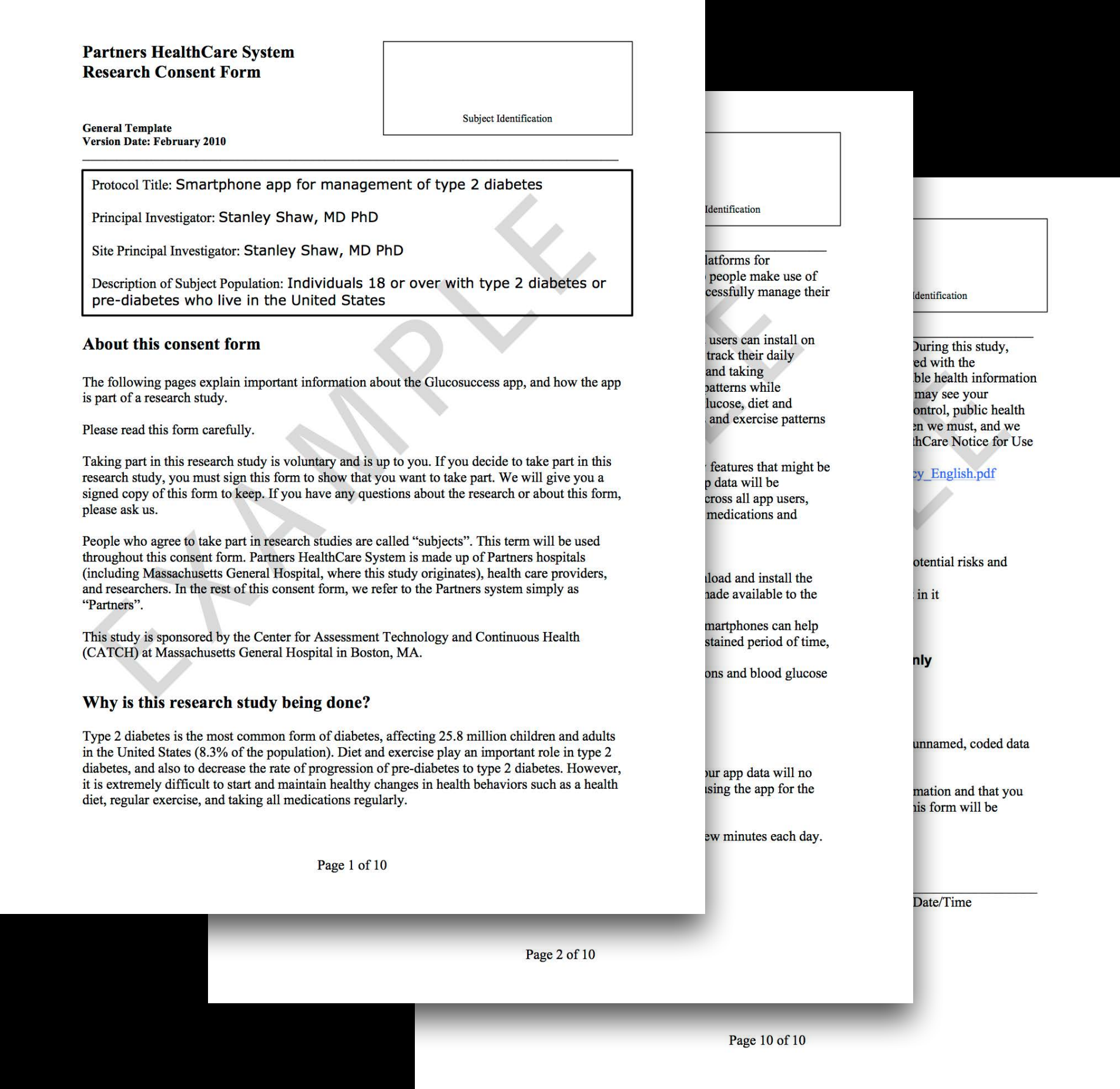
# Informed Consent

What is it?

What study entails

Risks and benefits

Often in person



# What is it?

## Risks and benefits

## From ethics review

Page 10 of 10

# What is it?

## Risks and benefits

## From ethics review

May need to sign

## Partners HealthCare System Research Consent Form

General Template  
Version Date: February 2010

Subject Identification

Protocol Title: Smartphone app for management of type 2 diabetes

Principal Investigator: Stanley Shaw, MD PhD

Site Principal Investigator: Stanley Shaw, MD PhD

Description of Subject Population: Individuals 18 or over with type 2 diabetes or pre-diabetes who live in the United States

### About this consent form

The following pages explain important information about the Glucosuccess app, and how the app is part of a research study.

Please read this form carefully.

Taking part in this research study is voluntary and is up to you. If you decide to take part in this research study, you must sign this form to show that you want to take part. We will give you a signed copy of this form to keep. If you have any questions about the research or about this form, please ask us.

People who agree to take part in research studies are called "subjects". This term will be used throughout this consent form. Partners HealthCare System is made up of Partners hospitals (including Massachusetts General Hospital, where this study originates), health care providers, and researchers. In the rest of this consent form, we refer to the Partners system simply as "Partners".

This study is sponsored by the Center for Assessment Technology and Continuous Health (CATCH) at Massachusetts General Hospital in Boston, MA.

### Why is this research study being done?

Type 2 diabetes is the most common form of diabetes, affecting 25.8 million children and adults in the United States (8.3% of the population). Diet and exercise play an important role in type 2 diabetes, and also to decrease the rate of progression of pre-diabetes to type 2 diabetes. However, it is extremely difficult to start and maintain healthy changes in health behaviors such as a health diet, regular exercise, and taking all medications regularly.

Page 1 of 10

Page 2 of 10

Page 10 of 10

# Informed Consent

# What is it?

# What study entails

## Risks and benefits

# Often in person

## From ethics review

May need to sign

Often a legal document

## Partners HealthCare System Research Consent Form

General Template  
Version Date: February 2010

Subject Identification

Protocol Title: Smartphone app for management of type 2 diabetes

Principal Investigator: Stanley Shaw, MD PhD

Site Principal Investigator: Stanley Shaw, MD PhD

Description of Subject Population: Individuals 18 or over with type 2 diabetes or pre-diabetes who live in the United States

### About this consent form

The following pages explain important information about the Glucosuccess app, and how the app is part of a research study.

Please read this form carefully.

Taking part in this research study is voluntary and is up to you. If you decide to take part in this research study, you must sign this form to show that you want to take part. We will give you a signed copy of this form to keep. If you have any questions about the research or about this form, please ask us.

People who agree to take part in research studies are called "subjects". This term will be used throughout this consent form. Partners HealthCare System is made up of Partners hospitals (including Massachusetts General Hospital, where this study originates), health care providers, and researchers. In the rest of this consent form, we refer to the Partners system simply as "Partners".

This study is sponsored by the Center for Assessment Technology and Continuous Health (CATCH) at Massachusetts General Hospital in Boston, MA.

### Why is this research study being done?

Type 2 diabetes is the most common form of diabetes, affecting 25.8 million children and adults in the United States (8.3% of the population). Diet and exercise play an important role in type 2 diabetes, and also to decrease the rate of progression of pre-diabetes to type 2 diabetes. However, it is extremely difficult to start and maintain healthy changes in health behaviors such as a health diet, regular exercise, and taking all medications regularly.

Page 1 of 10

Page 2 of 10

Page 10 of 10

# Consent Document

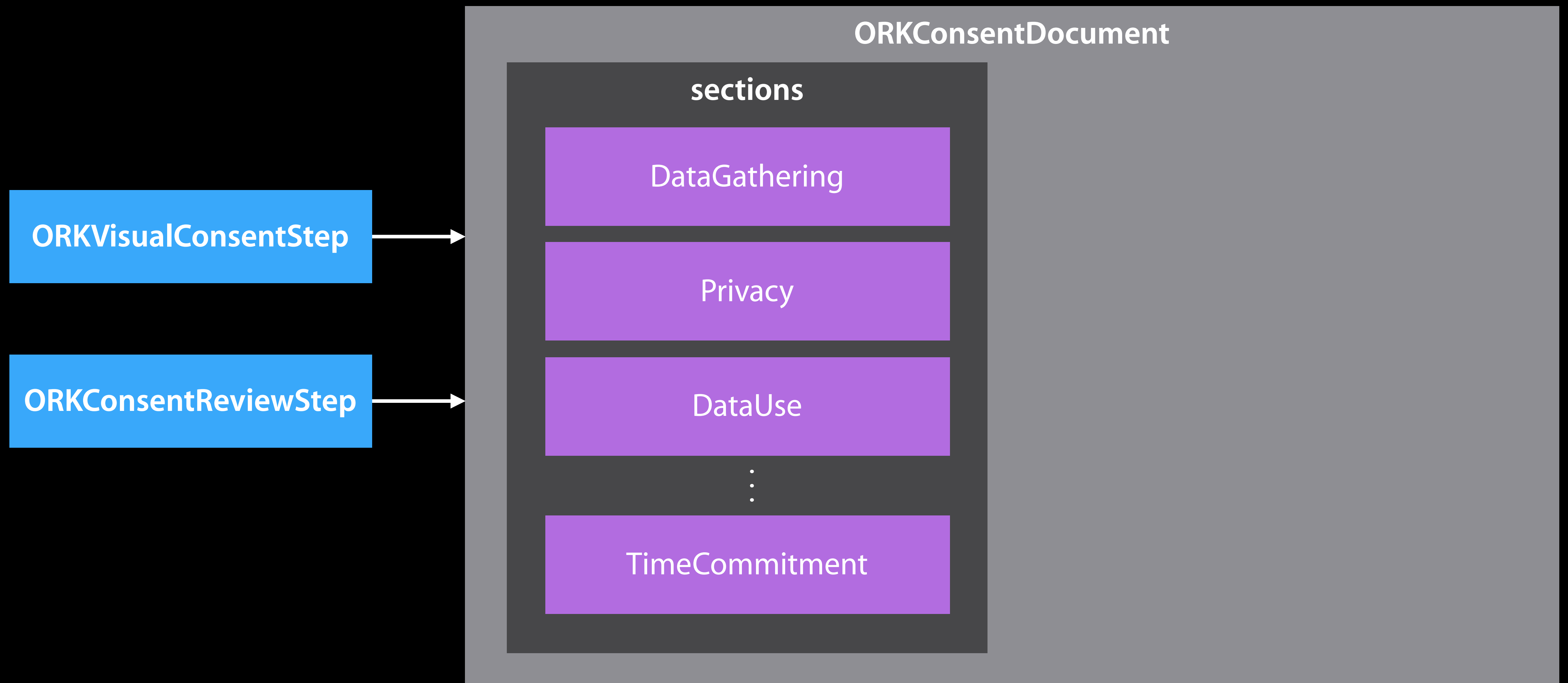
ORKVisualConsentStep

ORKConsentReviewStep

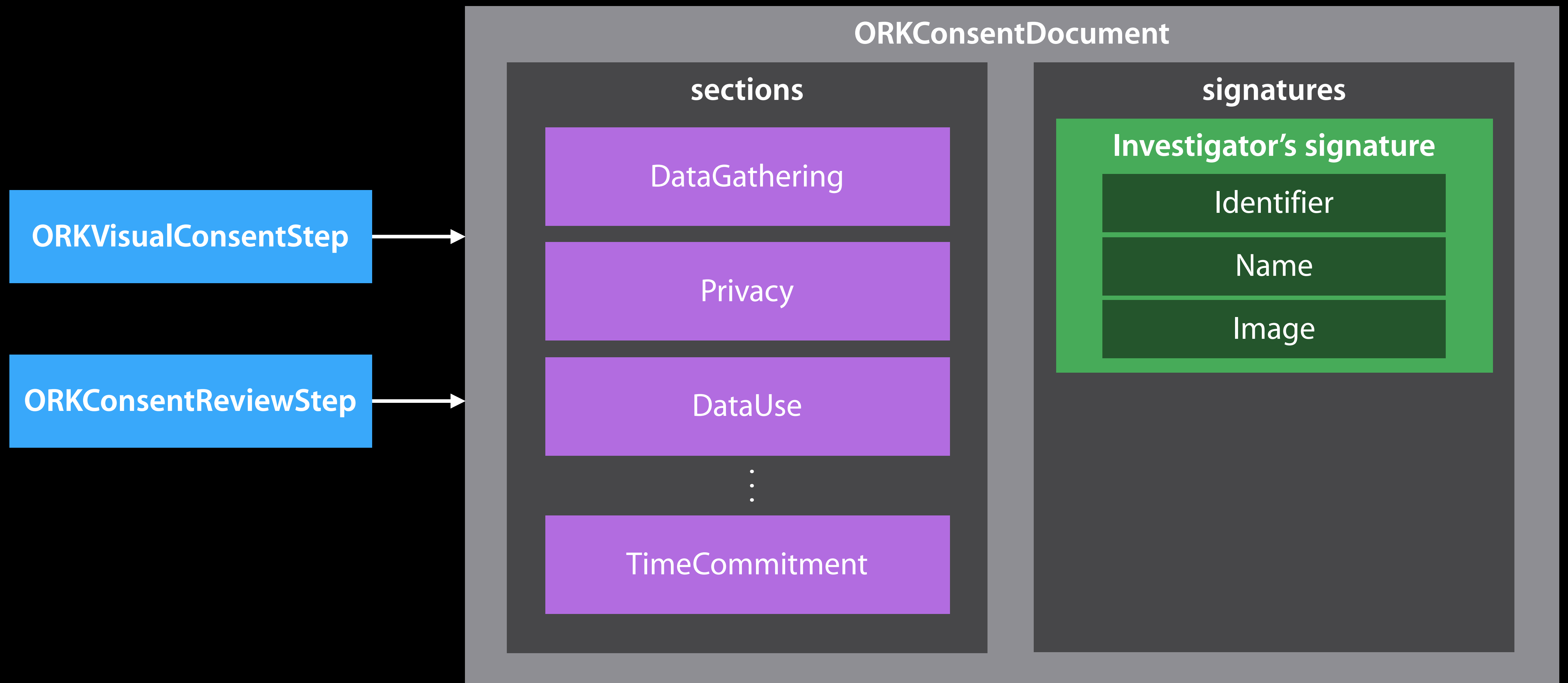
# Consent Document



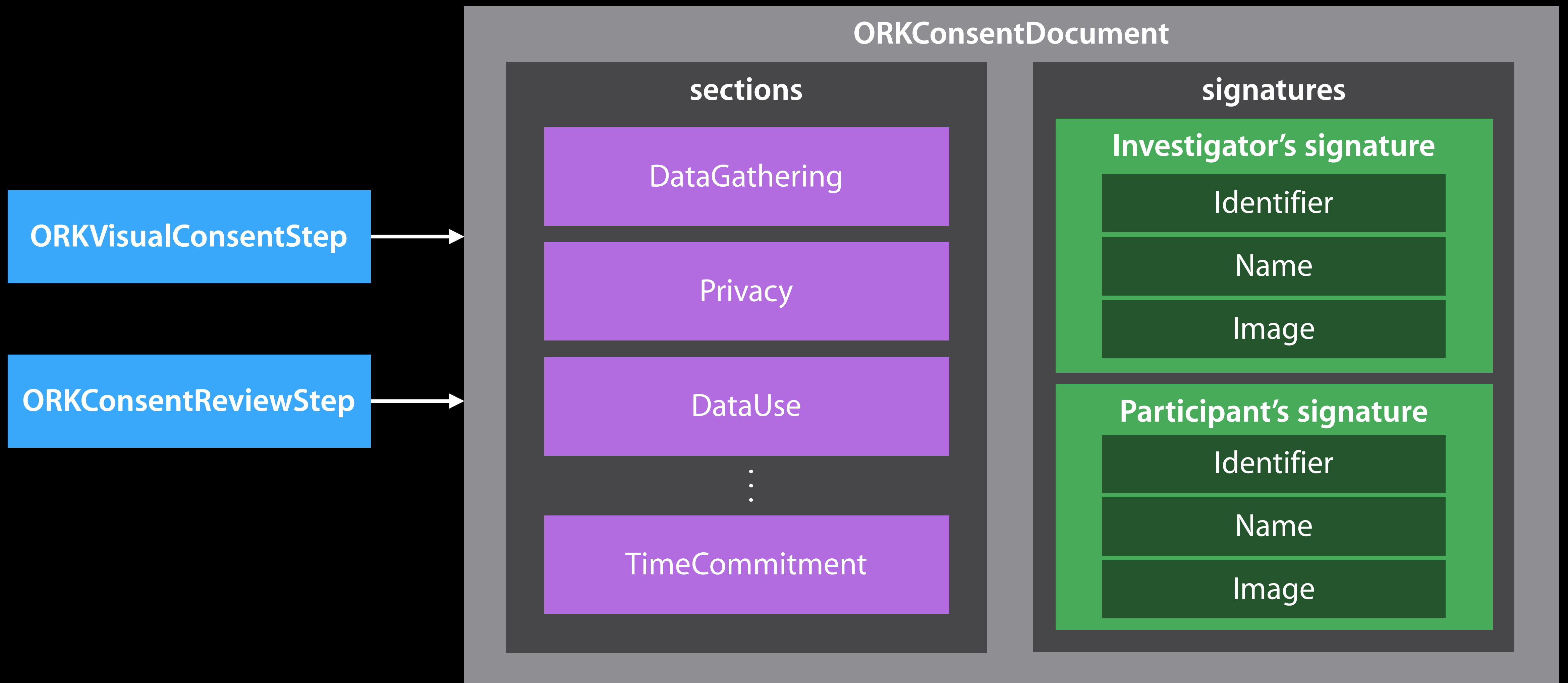
# Consent Document



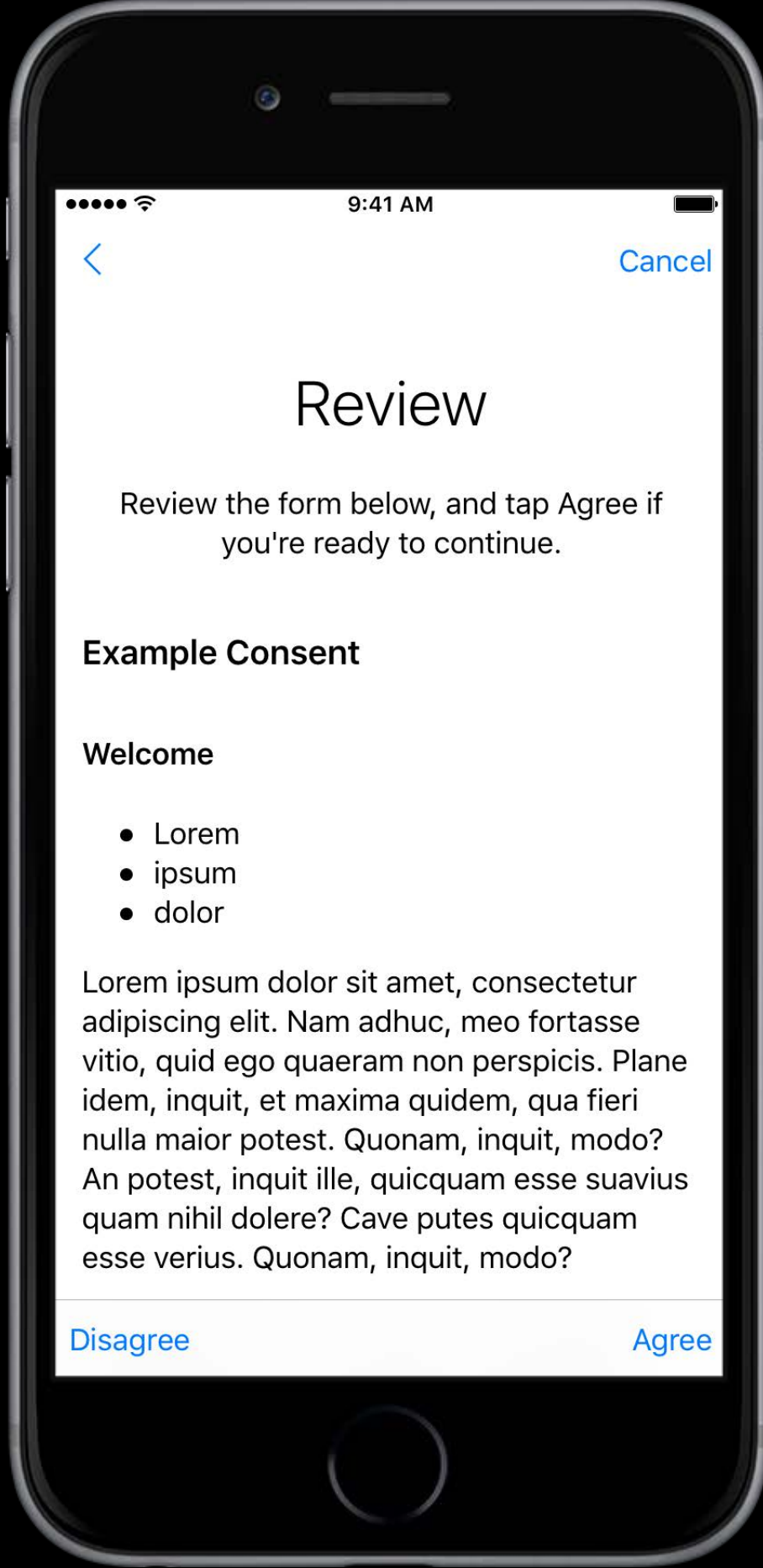
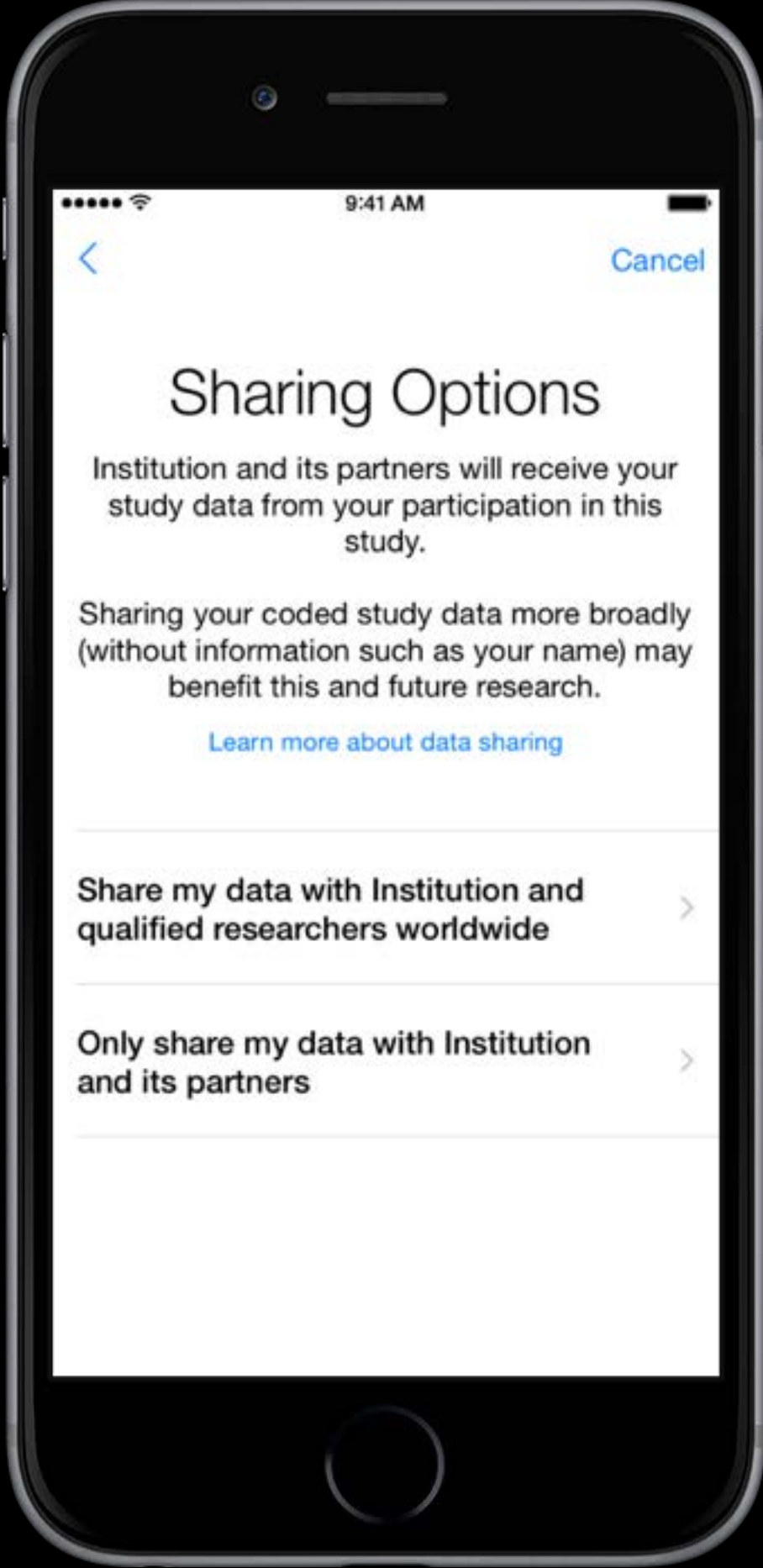
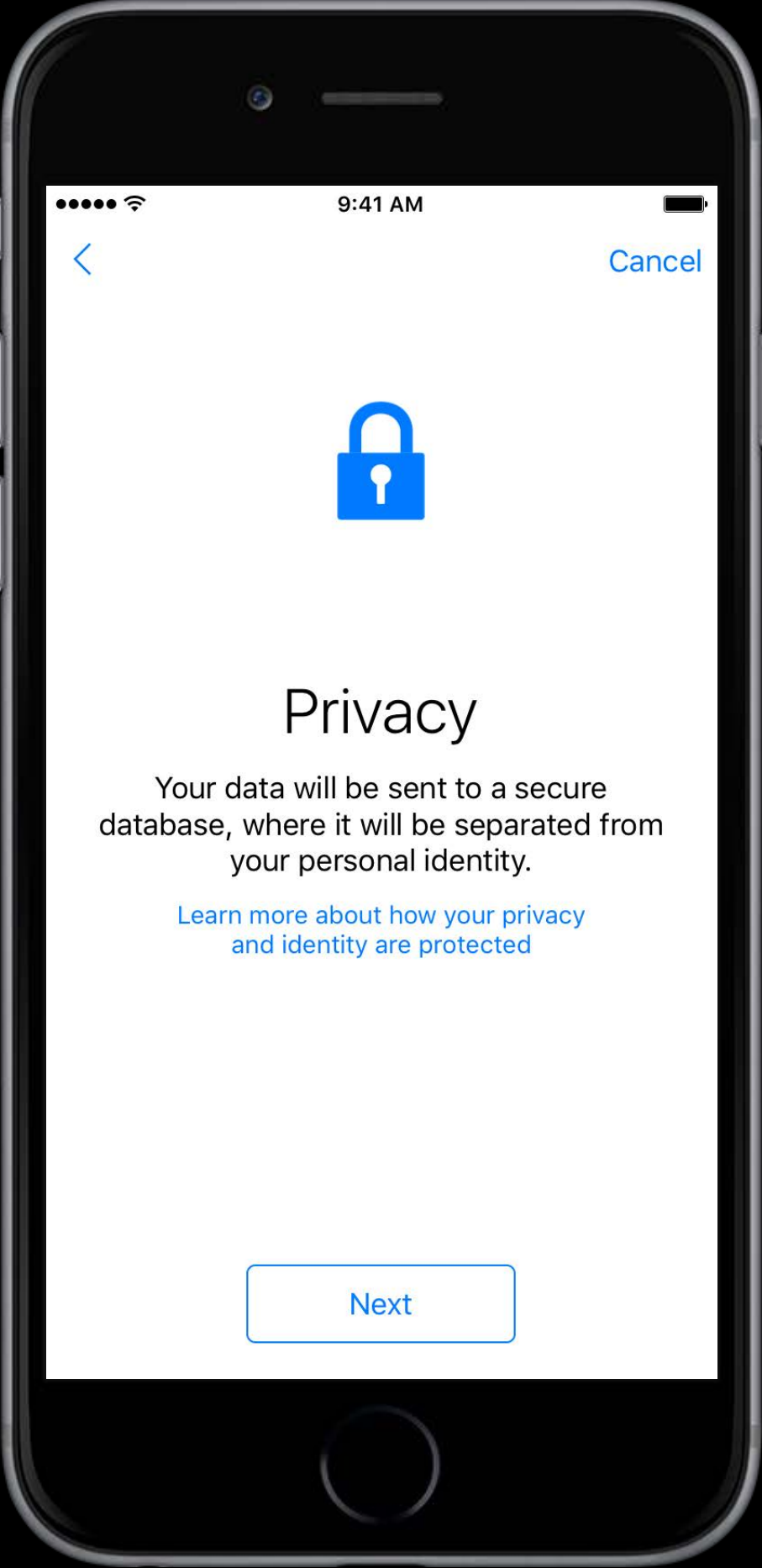
# Consent Document



# Consent Document



# Informed Consent Steps



# Visual Consent Step

One screen per section

Animated transitions

Customizable



# Visual Consent Step

```
// Create a section  
let consentSection = ORKConsentSection(type: .DataGathering)
```

# Visual Consent Step

```
// Create a section  
let consentSection = ORKConsentSection(type: .DataGathering)  
consentSection.title = "Sensor Data"
```

# Visual Consent Step

```
// Create a section  
let consentSection = ORKConsentSection(type: .DataGathering)  
consentSection.title = "Sensor Data"  
consentSection.summary = "We gather data using sensors ..."
```

# Visual Consent Step

```
// Create a section
let consentSection = ORKConsentSection(type: .DataGathering)
consentSection.title = "Sensor Data"
consentSection.summary = "We gather data using sensors ..."
consentSection.content = ... // "Learn More" content (+ `htmlContent`)
```

# Visual Consent Step

```
// Create a section
let consentSection = ORKConsentSection(type: .DataGathering)
consentSection.title = "Sensor Data"
consentSection.summary = "We gather data using sensors ..."
consentSection.content = ... // "Learn More" content (+ `htmlContent`)
consentSection.customImage = UIImage(named: "CustomImage")
```

# Visual Consent Step

```
// Create a section
let consentSection = ORKConsentSection(type: .DataGathering)
consentSection.title = "Sensor Data"
consentSection.summary = "We gather data using sensors ..."
consentSection.content = ... // "Learn More" content (+ `htmlContent`)
consentSection.customImage = UIImage(named: "CustomImage")
consentSection.customAnimationURL = ... // File URL to video for transition
```

# Visual Consent Step

```
// Create a section
let consentSection = ORKConsentSection(type: .DataGathering)
consentSection.title = "Sensor Data"
consentSection.summary = "We gather data using sensors ..."
consentSection.content = ... // "Learn More" content (+ `htmlContent`)
consentSection.customImage = UIImage(named: "CustomImage")
consentSection.customAnimationURL = ... // File URL to video for transition

let consentDocument = ORKConsentDocument()
consentDocument.sections = [consentSection, ...] // Attach to a document
```

# Visual Consent Step

```
// Create a section
let consentSection = ORKConsentSection(type: .DataGathering)
consentSection.title = "Sensor Data"
consentSection.summary = "We gather data using sensors ..."
consentSection.content = ... // "Learn More" content (+ `htmlContent`)
consentSection.customImage = UIImage(named: "CustomImage")
consentSection.customAnimationURL = ... // File URL to video for transition

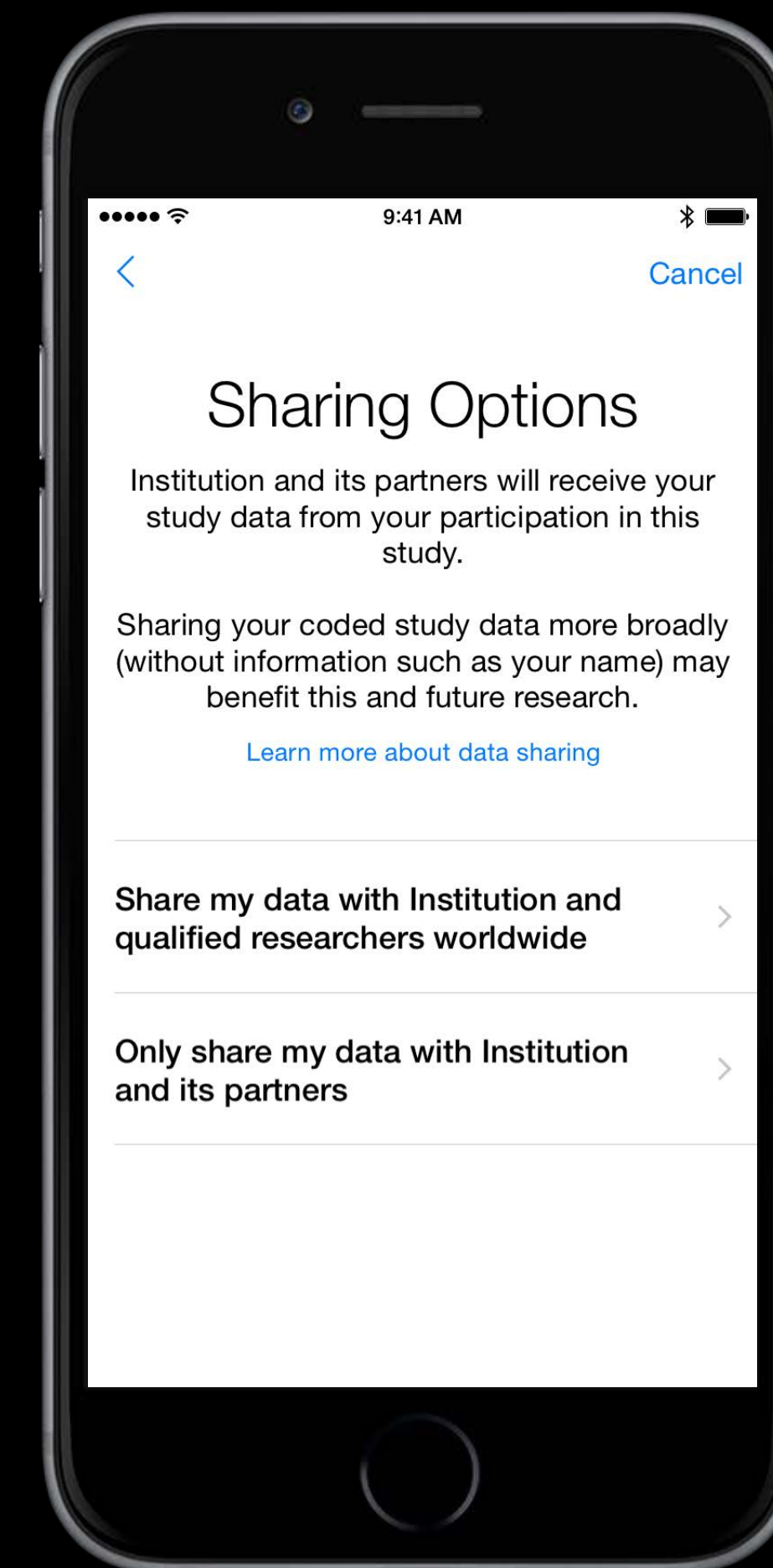
let consentDocument = ORKConsentDocument()
consentDocument.sections = [consentSection, ...] // Attach to a document

let visualConsentStep =
    ORKVisualConsentStep(identifier: "visualConsent",
        document: consentDocument)
```

# Sharing Consent Step

Give participants control

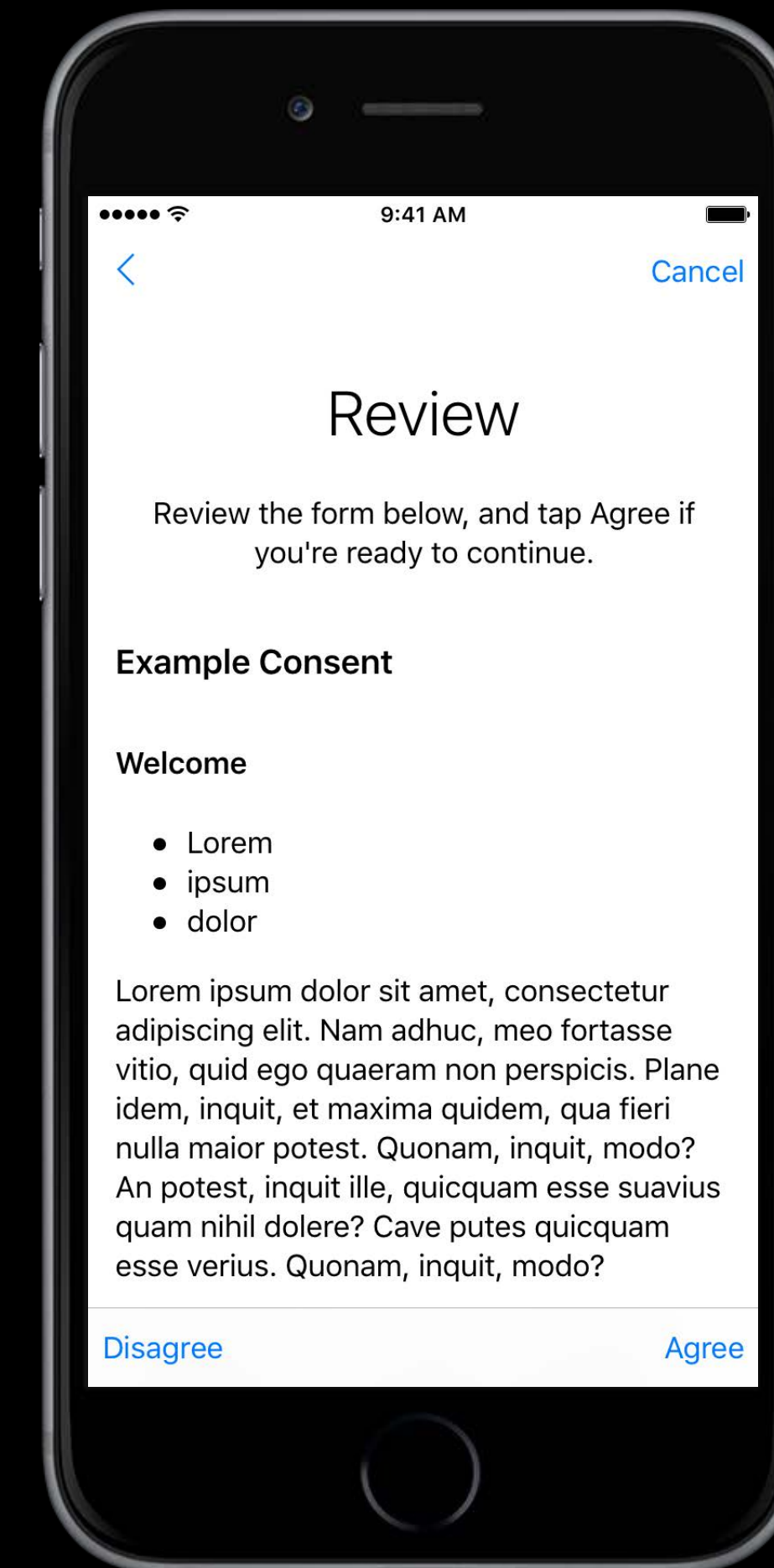
- ORKConsentSharingStep
- Supply institution name
- Localized



# Consent Review Step

## Participant

- Reviews document
- Enters name (optional)
- Signs with touch (optional)



# Consent Review Step

```
let consentDocument = ...    // Our consent document from before
```

# Consent Review Step

```
let consentDocument = ... // Our consent document from before
let participantSignature =
    ORKConsentSignature(forPersonWithTitle: "Participant",
        dateFormatString: nil,
        identifier: "participant")
```

# Consent Review Step

```
let consentDocument = ... // Our consent document from before
let participantSignature =
    ORKConsentSignature(forPersonWithTitle: "Participant",
        dateFormatString: nil,
        identifier: "participant")
participantSignature.requiresSignatureImage = false
```

# Consent Review Step

```
let consentDocument = ... // Our consent document from before
let participantSignature =
    ORKConsentSignature(forPersonWithTitle: "Participant",
        dateFormatString: nil,
        identifier: "participant")
participantSignature.requiresSignatureImage = false
consentDocument.addSignature(participantSignature)
```

# Consent Review Step

```
let consentDocument = ... // Our consent document from before
let participantSignature =
    ORKConsentSignature(forPersonWithTitle: "Participant",
        dateFormatString: nil,
        identifier: "participant")
participantSignature.requiresSignatureImage = false
consentDocument.addSignature(participantSignature)

let consentReviewStep =
    ORKConsentReviewStep(identifier: "consentReview",
        signature: signature,
        inDocument: consentDocument)
```

*Demo*

Consent

# Informed Consent

Things to consider

# Informed Consent

## Things to consider

“Informed”

---

Form factor

Custom sections

Questions from participants

Comprehension quiz

# Informed Consent

## Things to consider

### “Informed”

Form factor

Custom sections

Questions from participants

Comprehension quiz

### “Consent”

Verifying identity

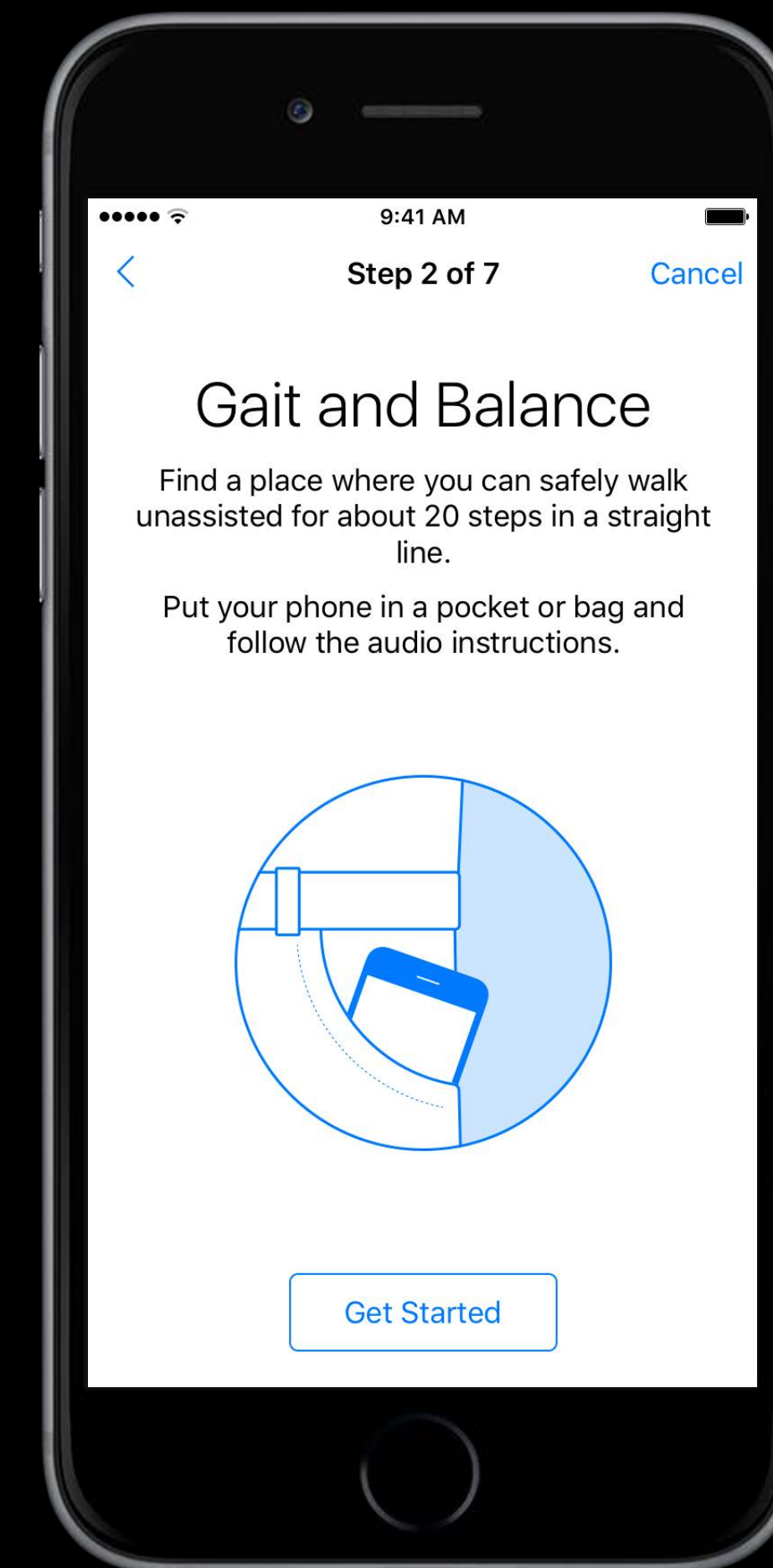
Digital signatures

# Active Tasks

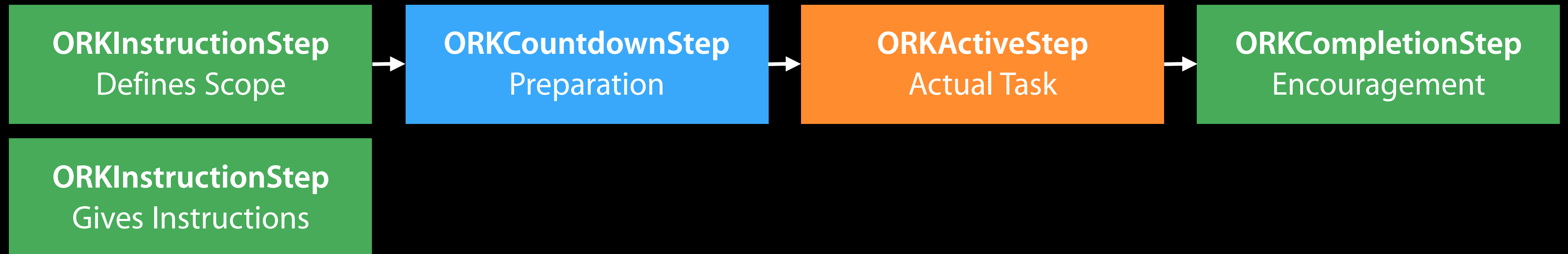
Interactive

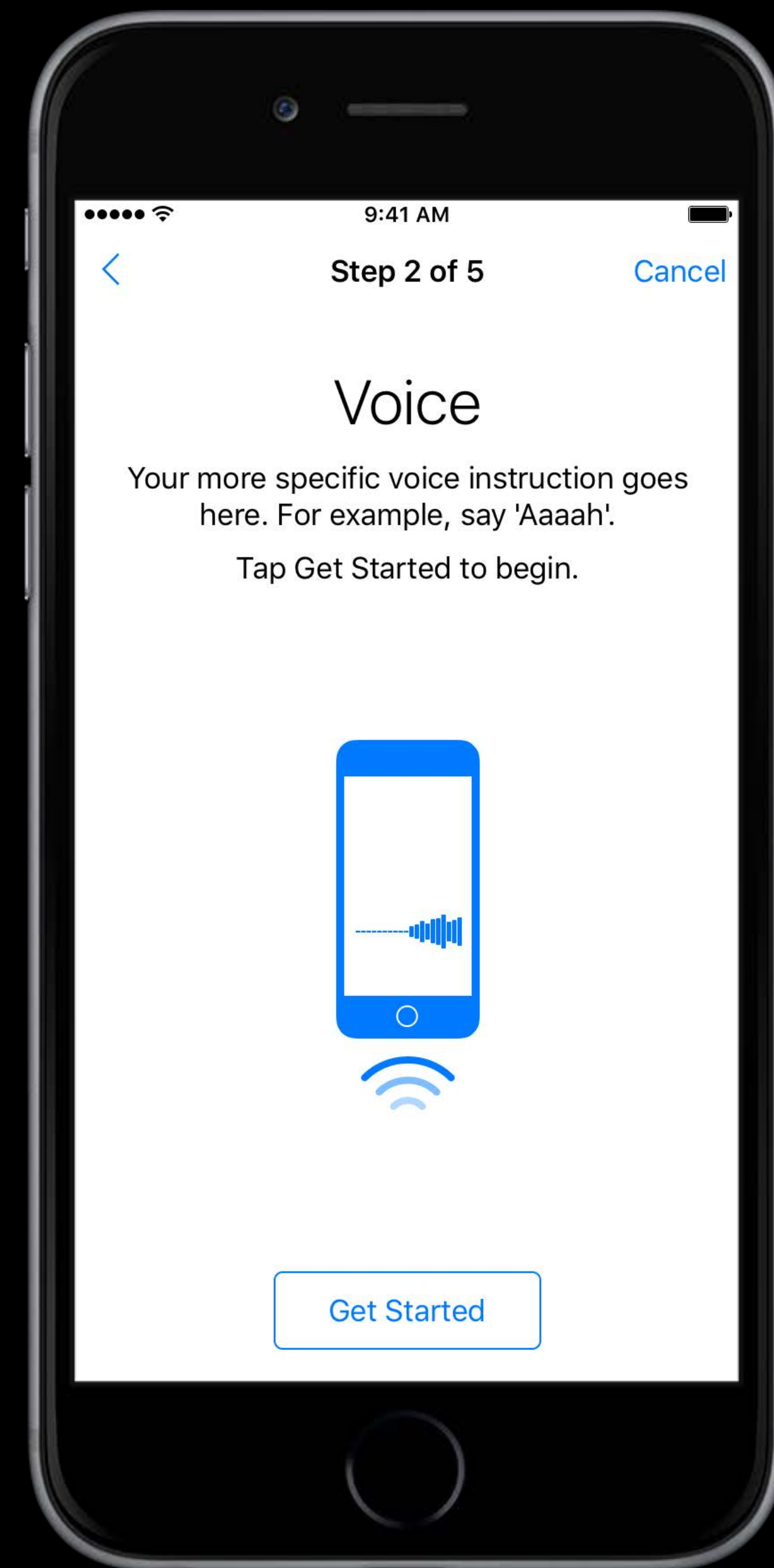
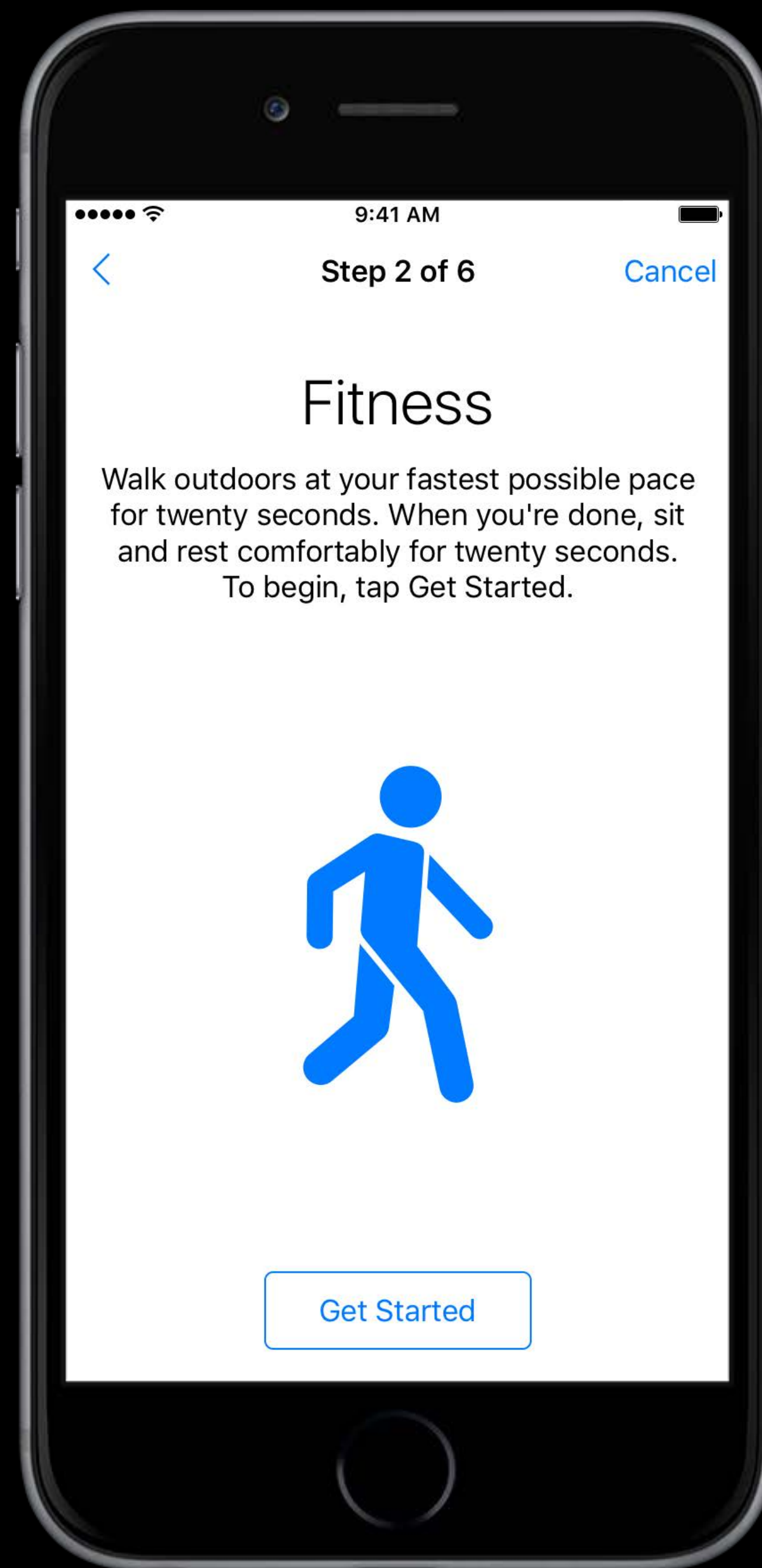
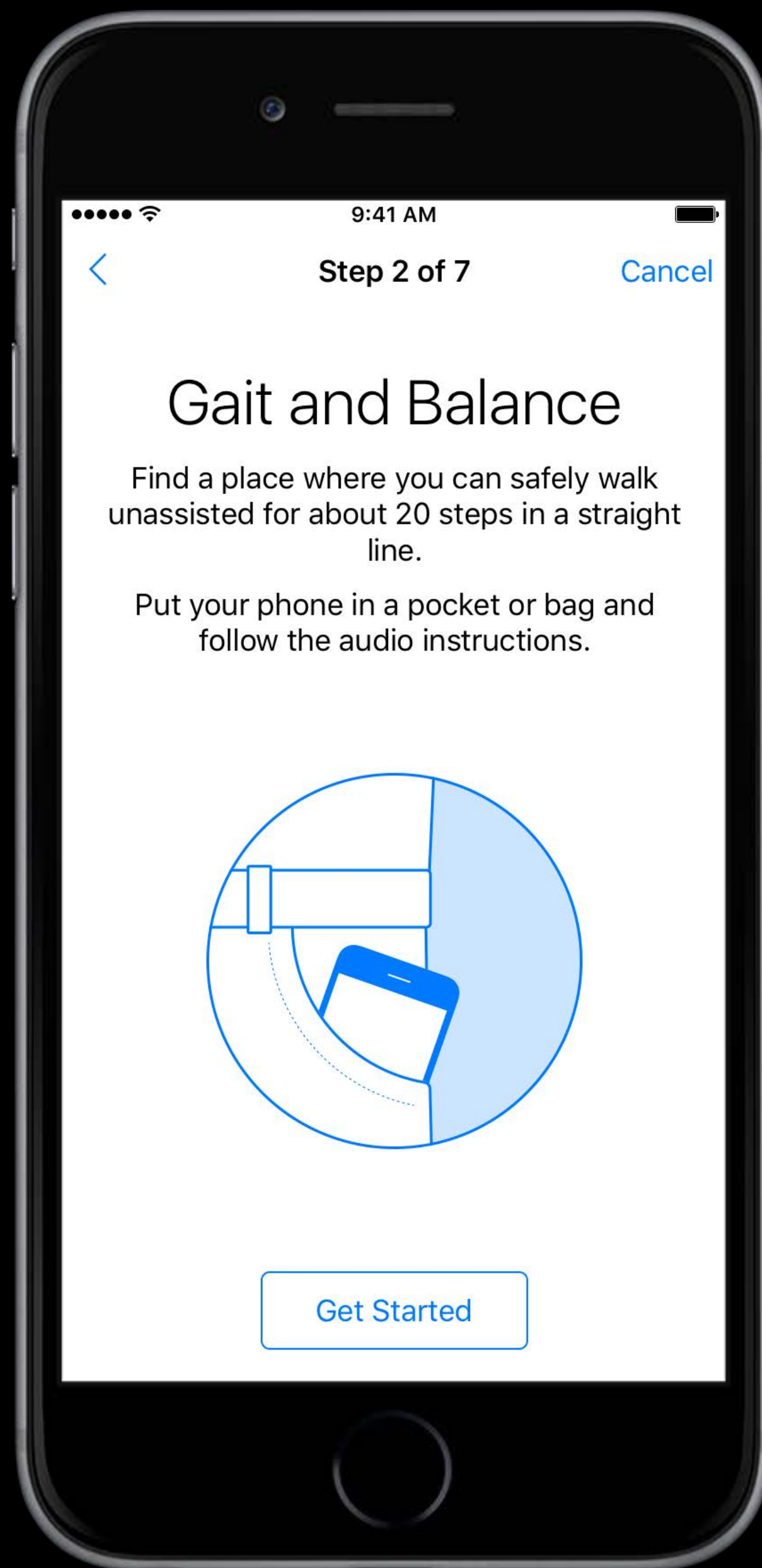
Short

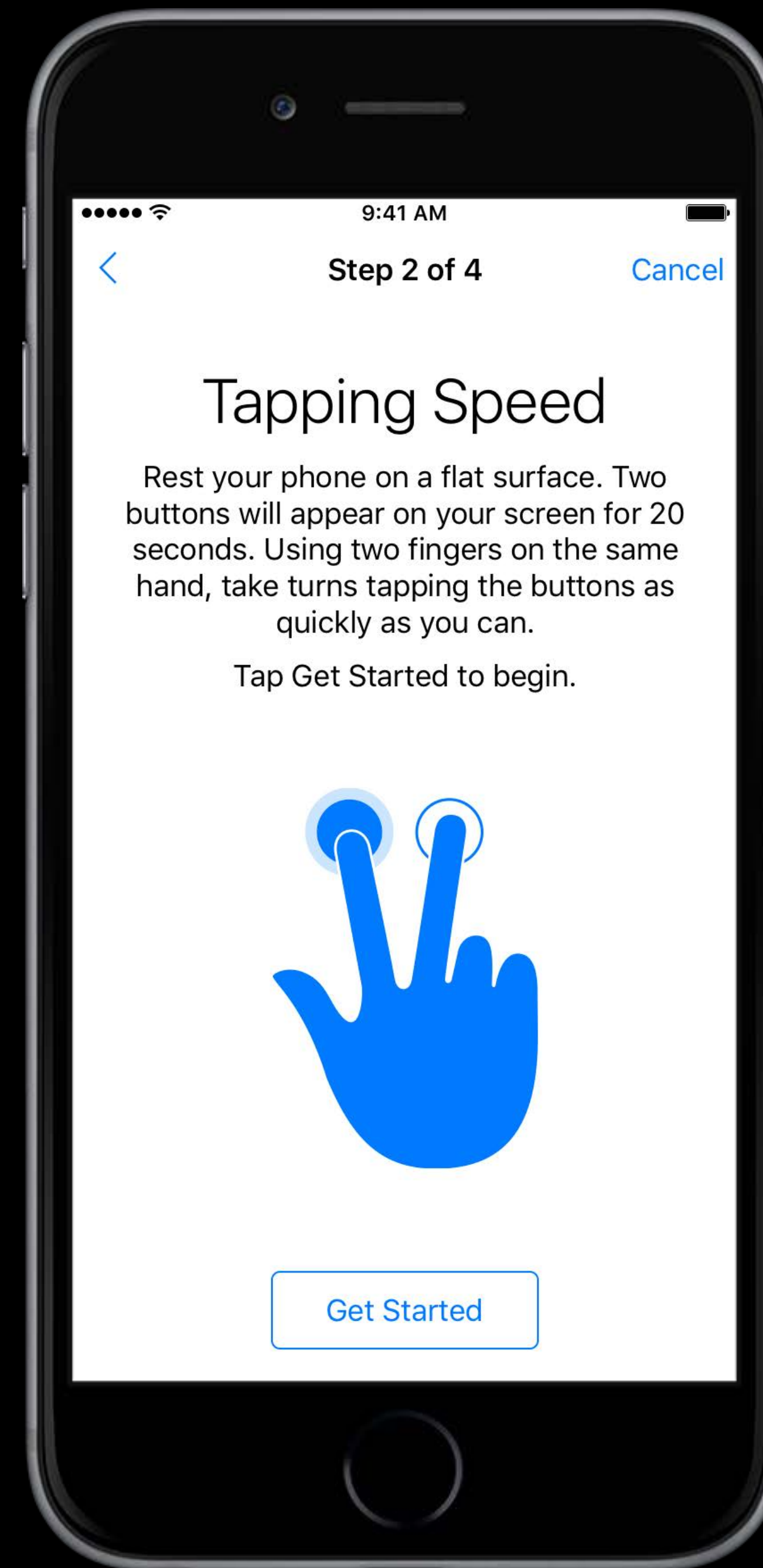
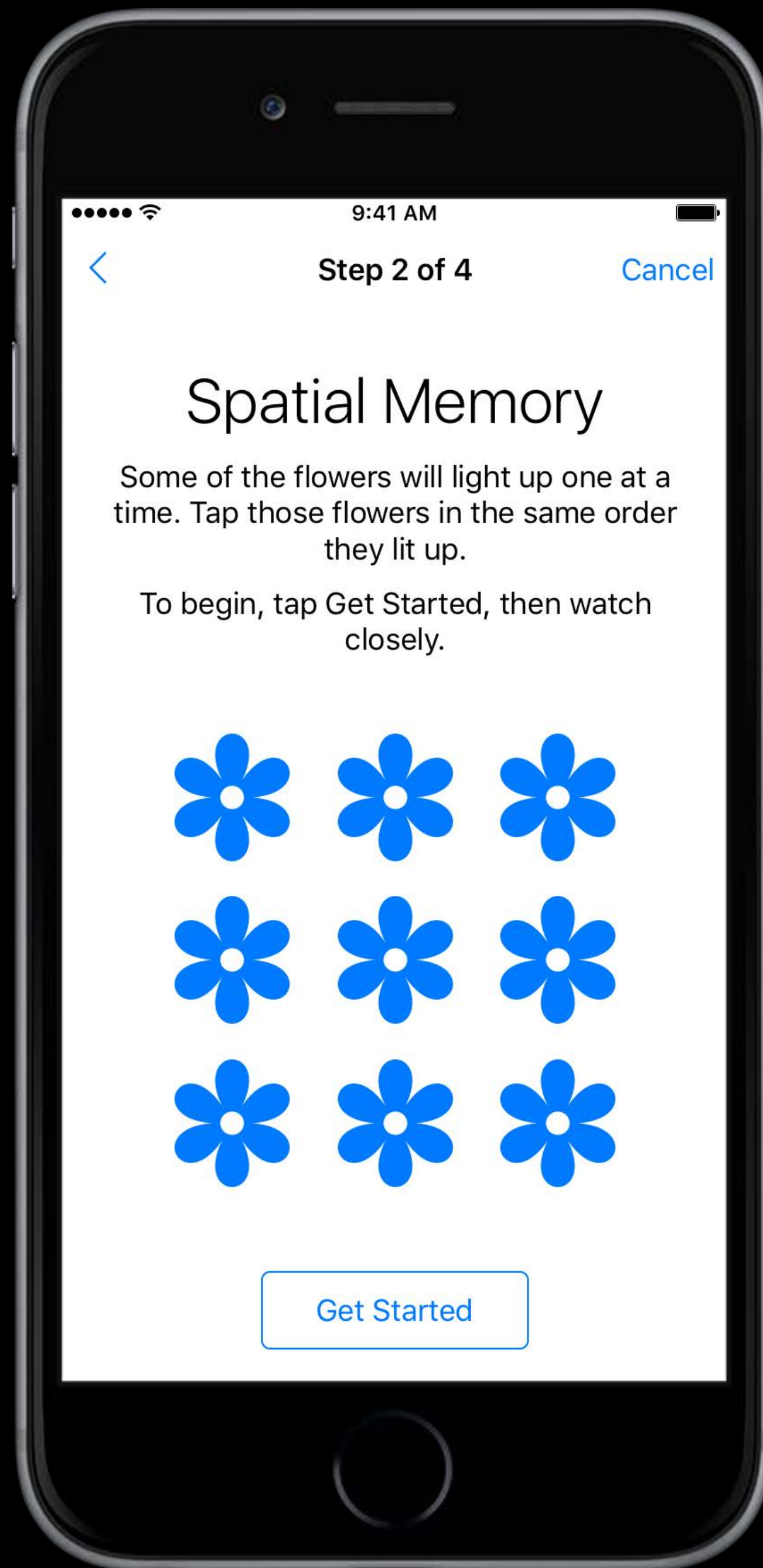
Use Sensors

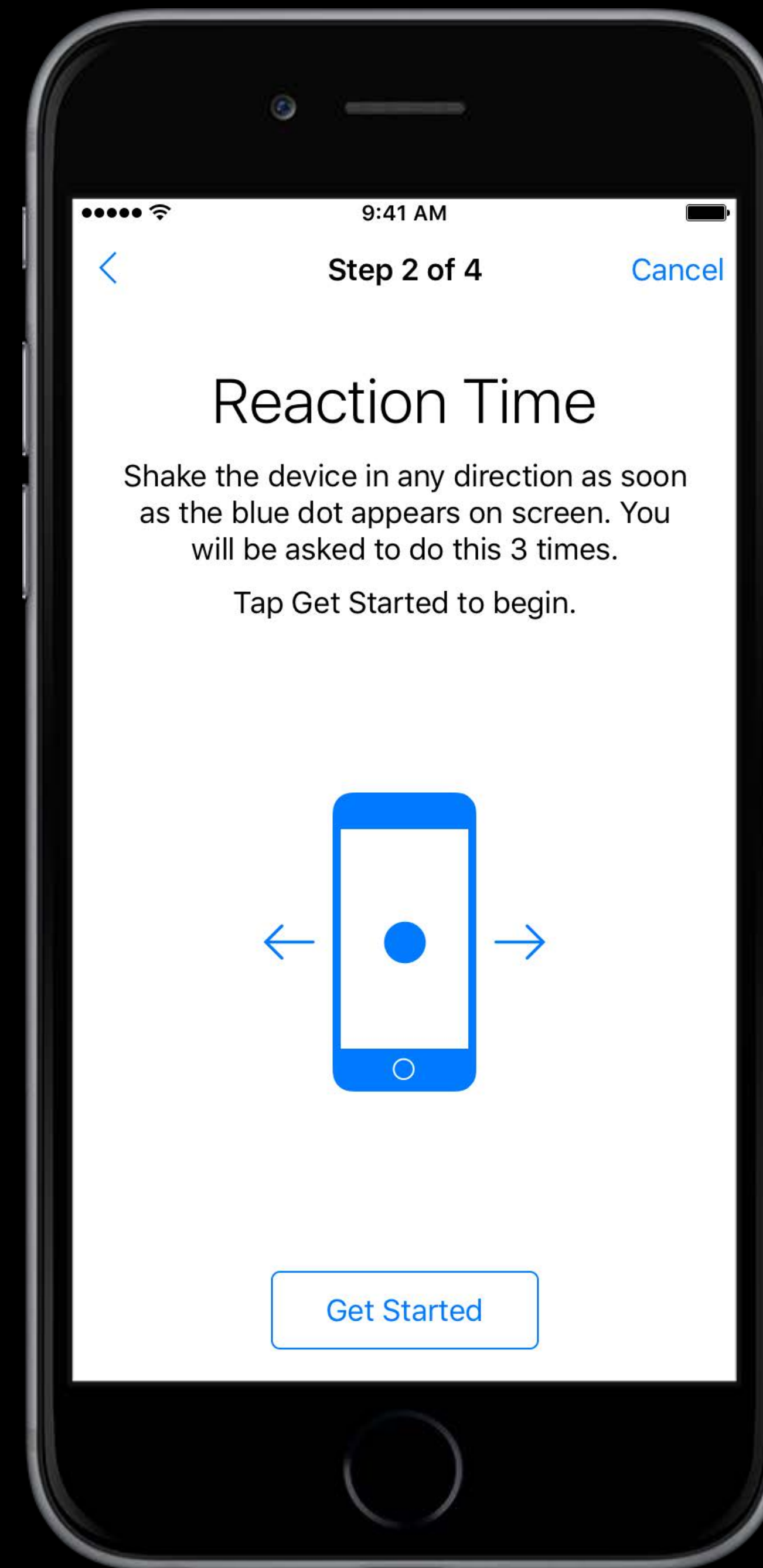
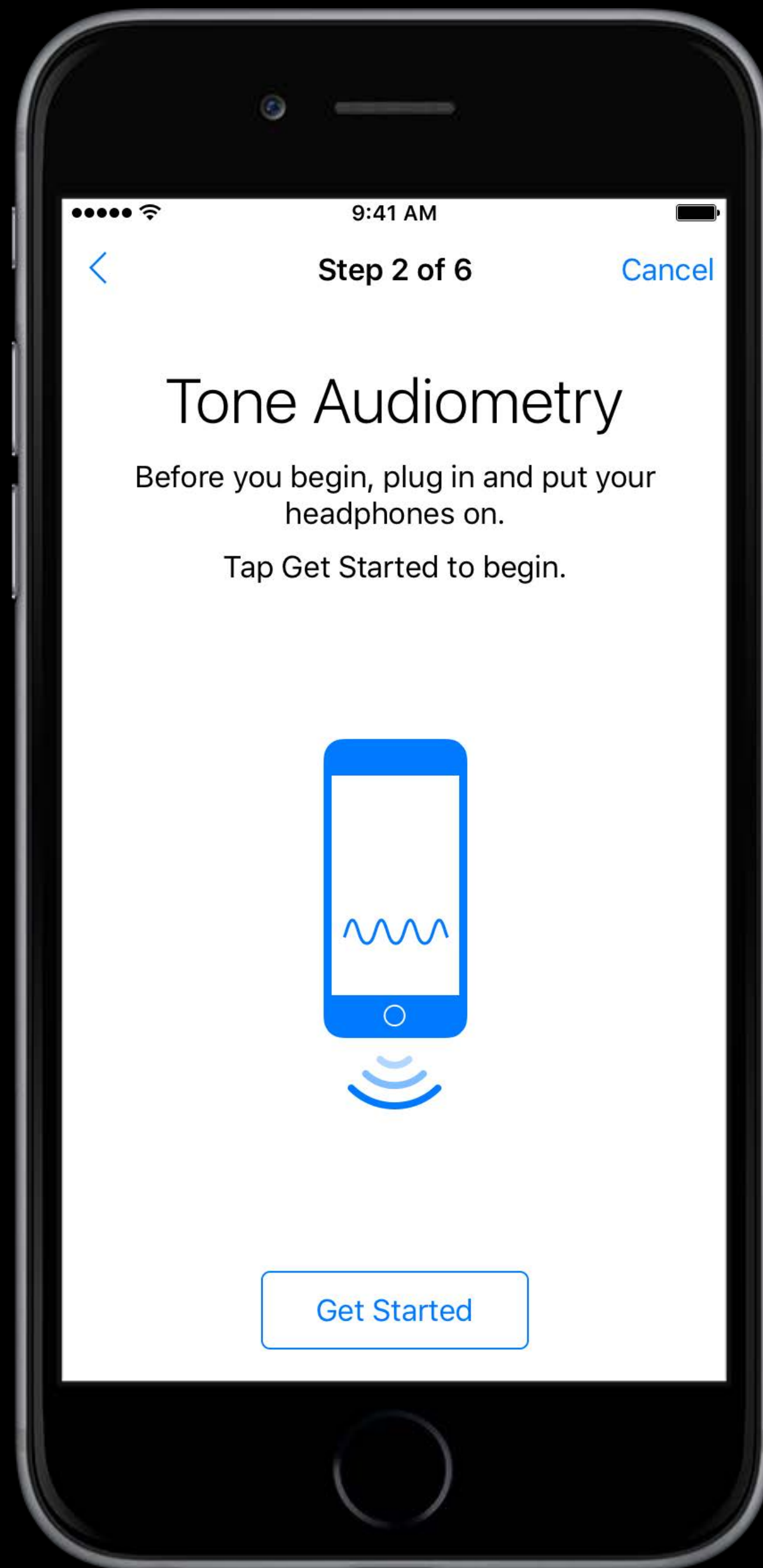


# Active Task Structure









*Demo*

Active Tasks

Contributing to ResearchKit

# ResearchKit Community

Pick or open an issue

Submit a pull request

Work with review team

Change gets merged



# Contributions

Active Tasks

Answer formats

Steps

Device support

Back end integrations

# Contributions

## Active Tasks

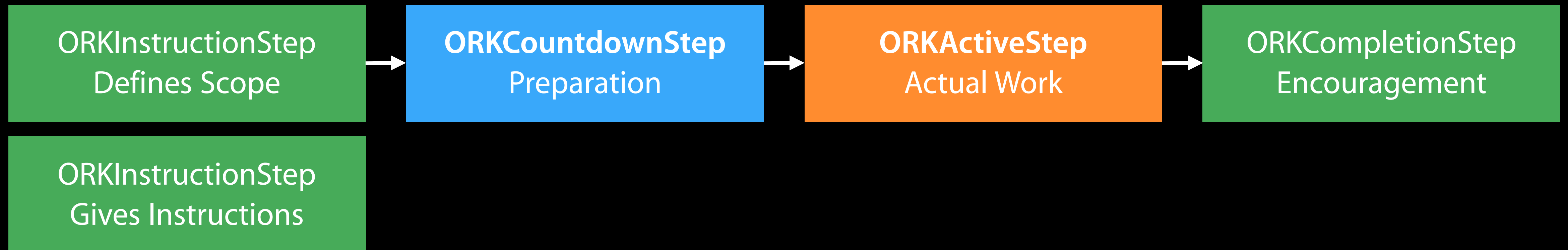
Answer formats

Steps

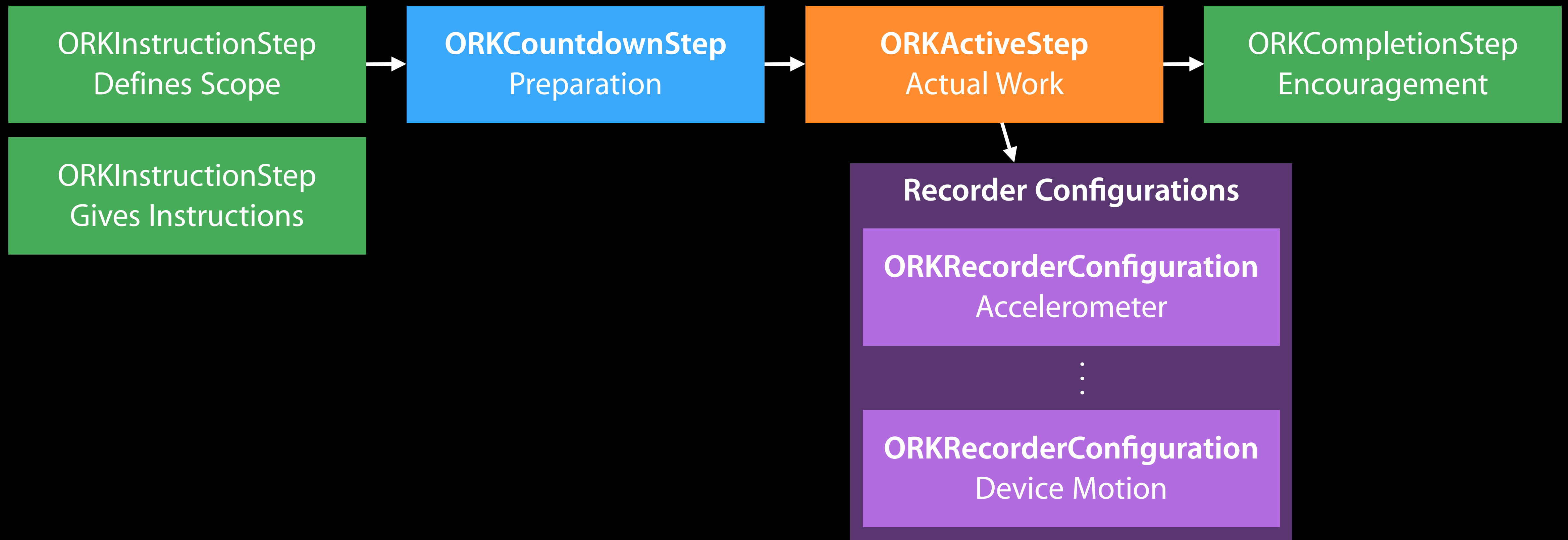
Device support

Back end integrations

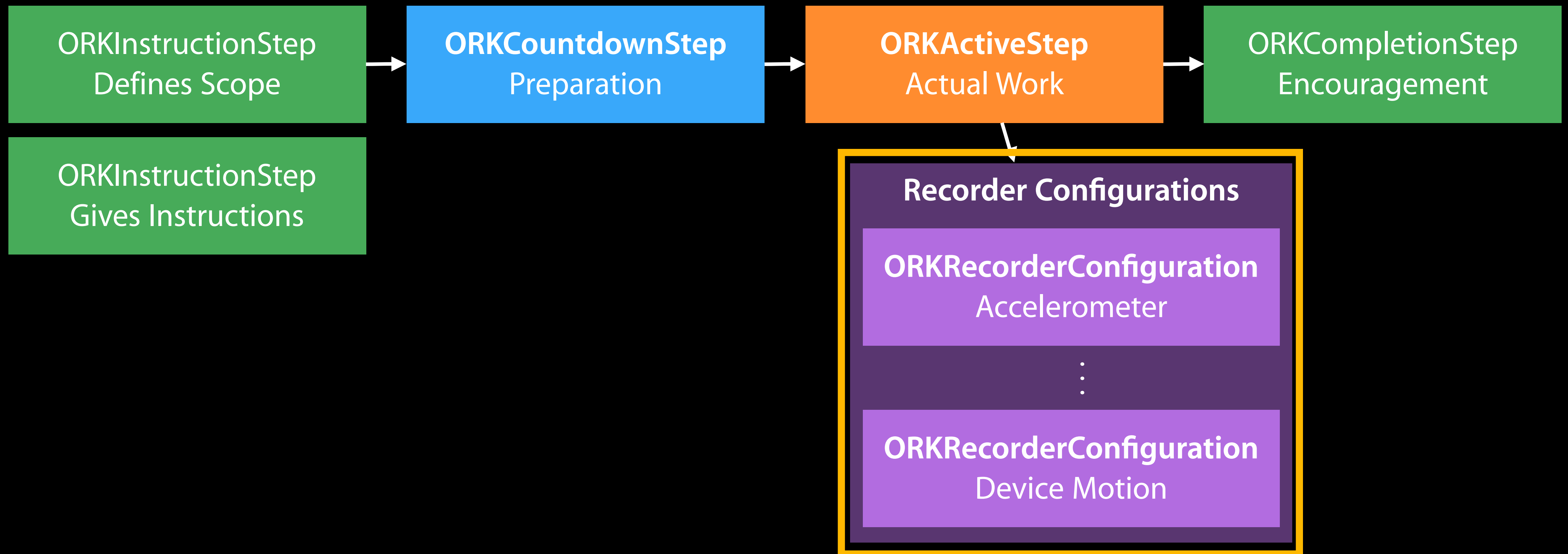
# Active Tasks



# Active Tasks



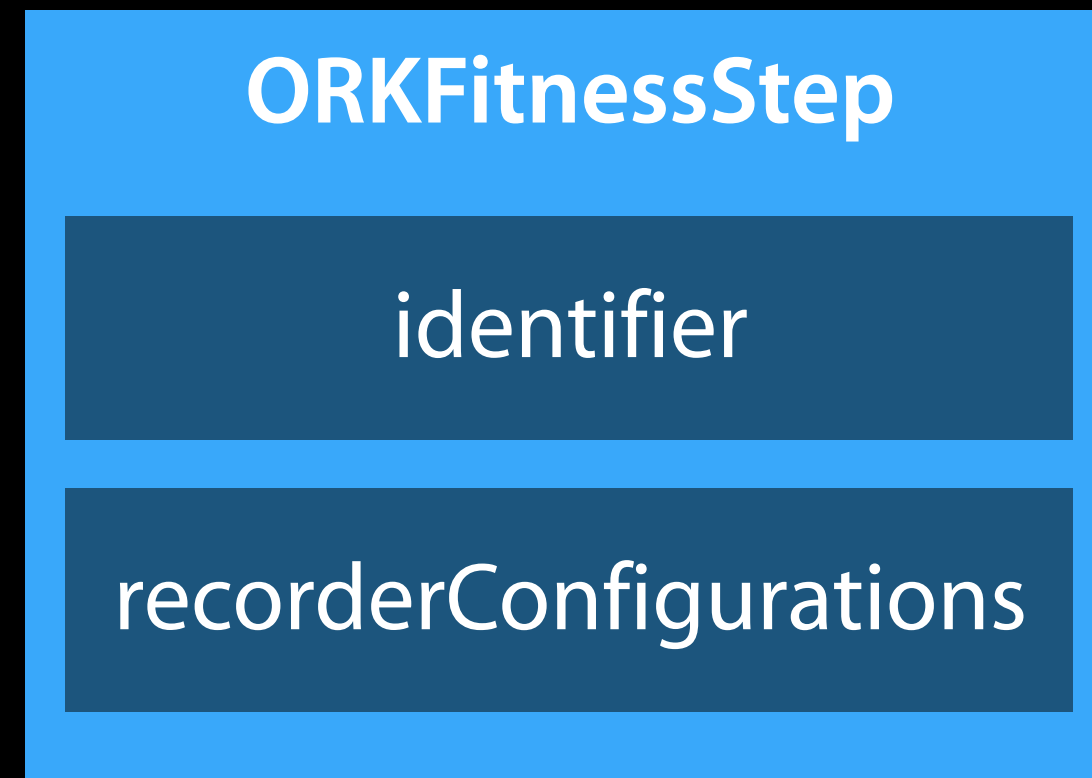
# Active Tasks



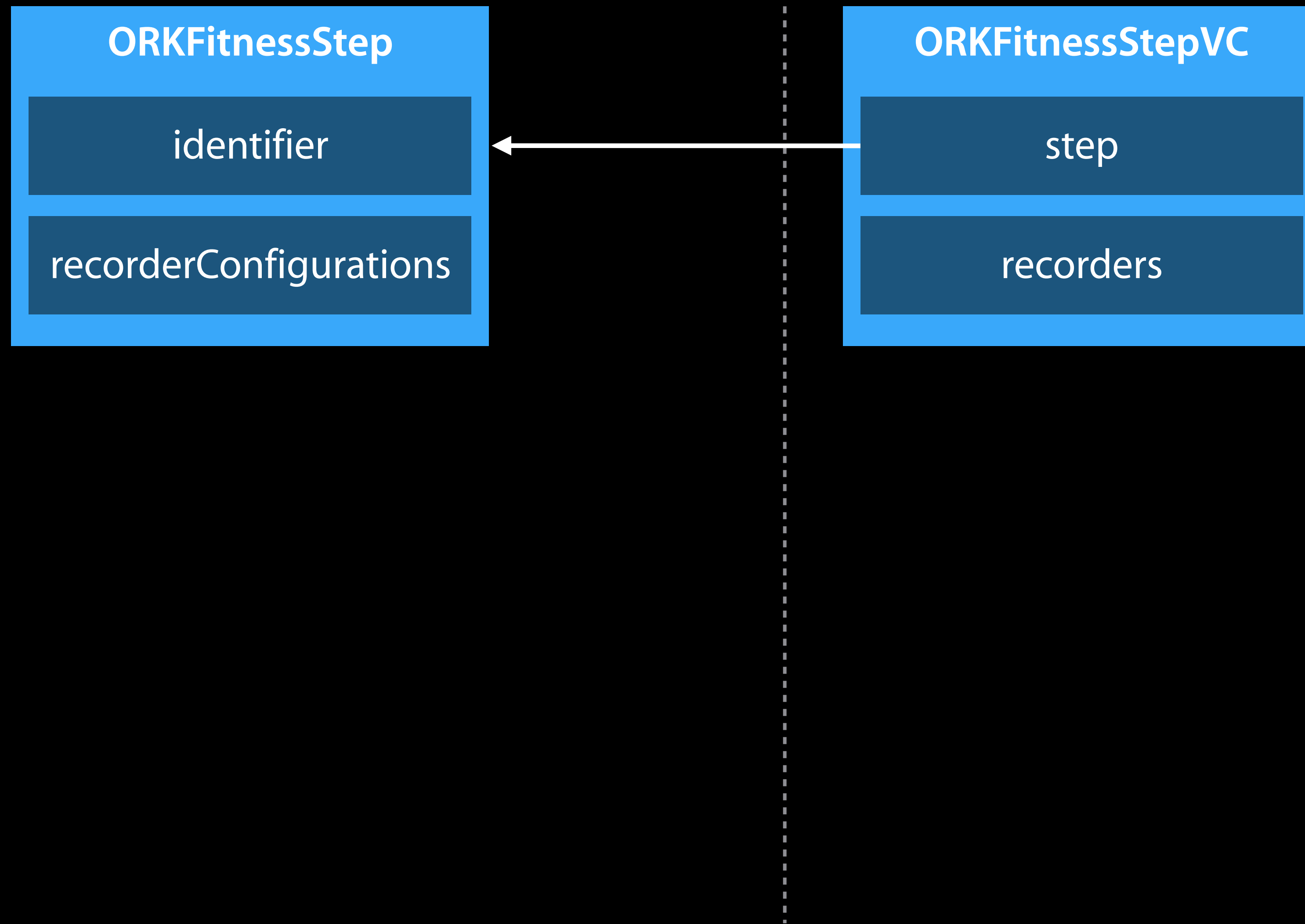
# Recorders

Recorder	Data Collected	Parameters
ORKAccelerometerRecorder	CMAccelerometerData	frequency
ORKDeviceMotionRecorder	CMDeviceMotion	frequency
ORKHealthQuantityTypeRecorder	HKQuantitySample	quantityType, unit
ORKLocationRecorder	CLLocation	N/A
ORKPedometerRecorder	CMPedometerData	N/A

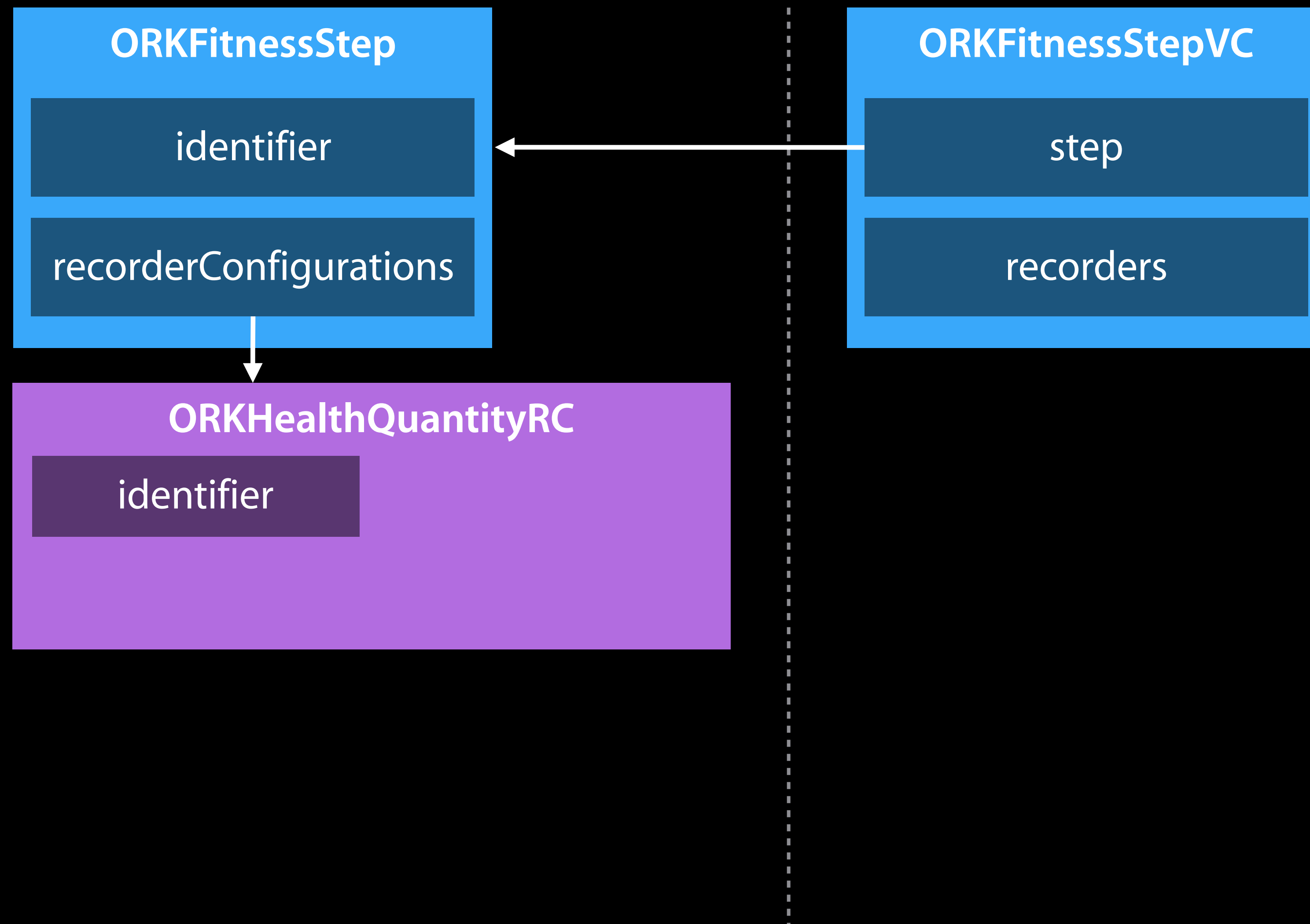
# Using Recorders



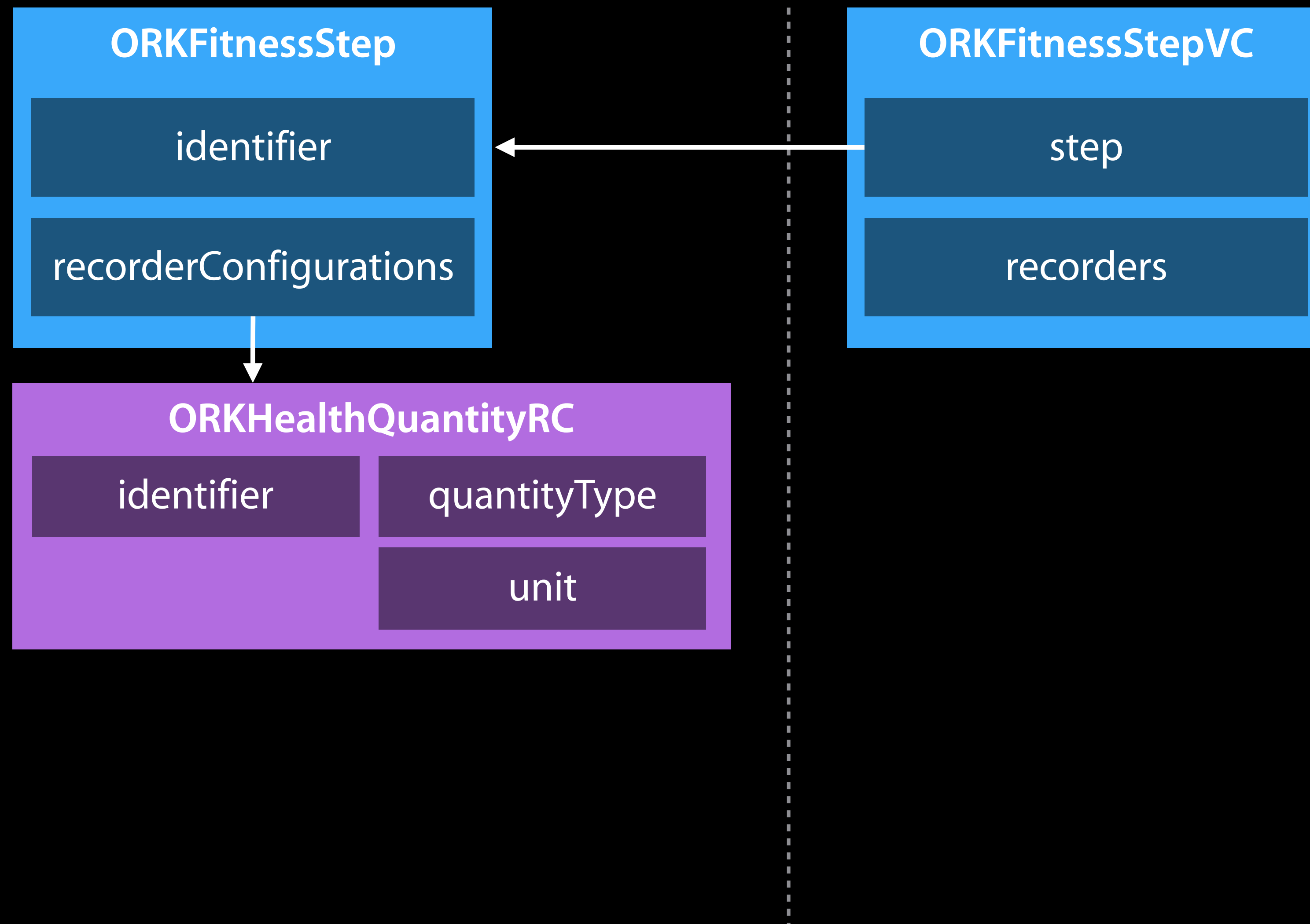
# Using Recorders



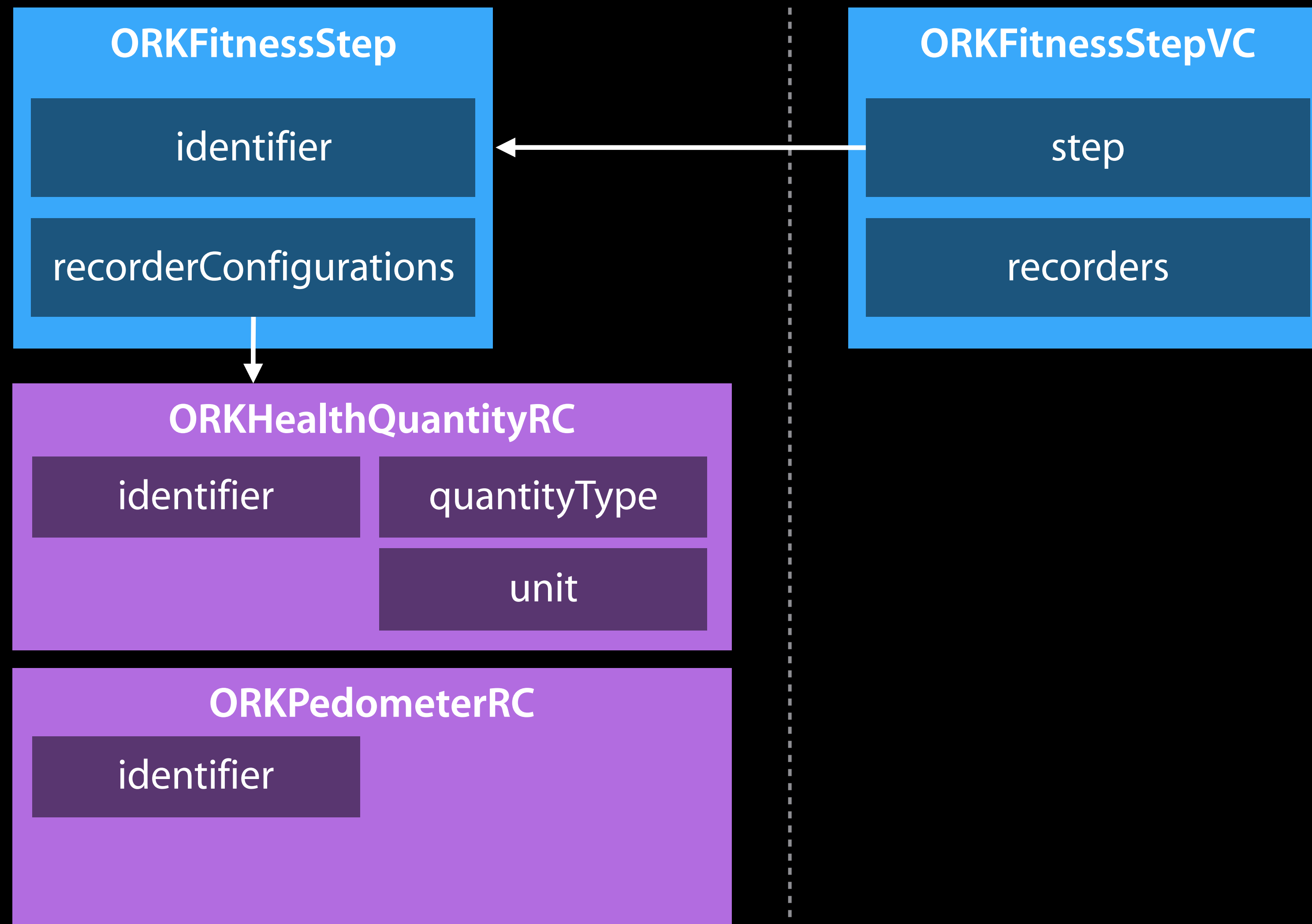
# Using Recorders



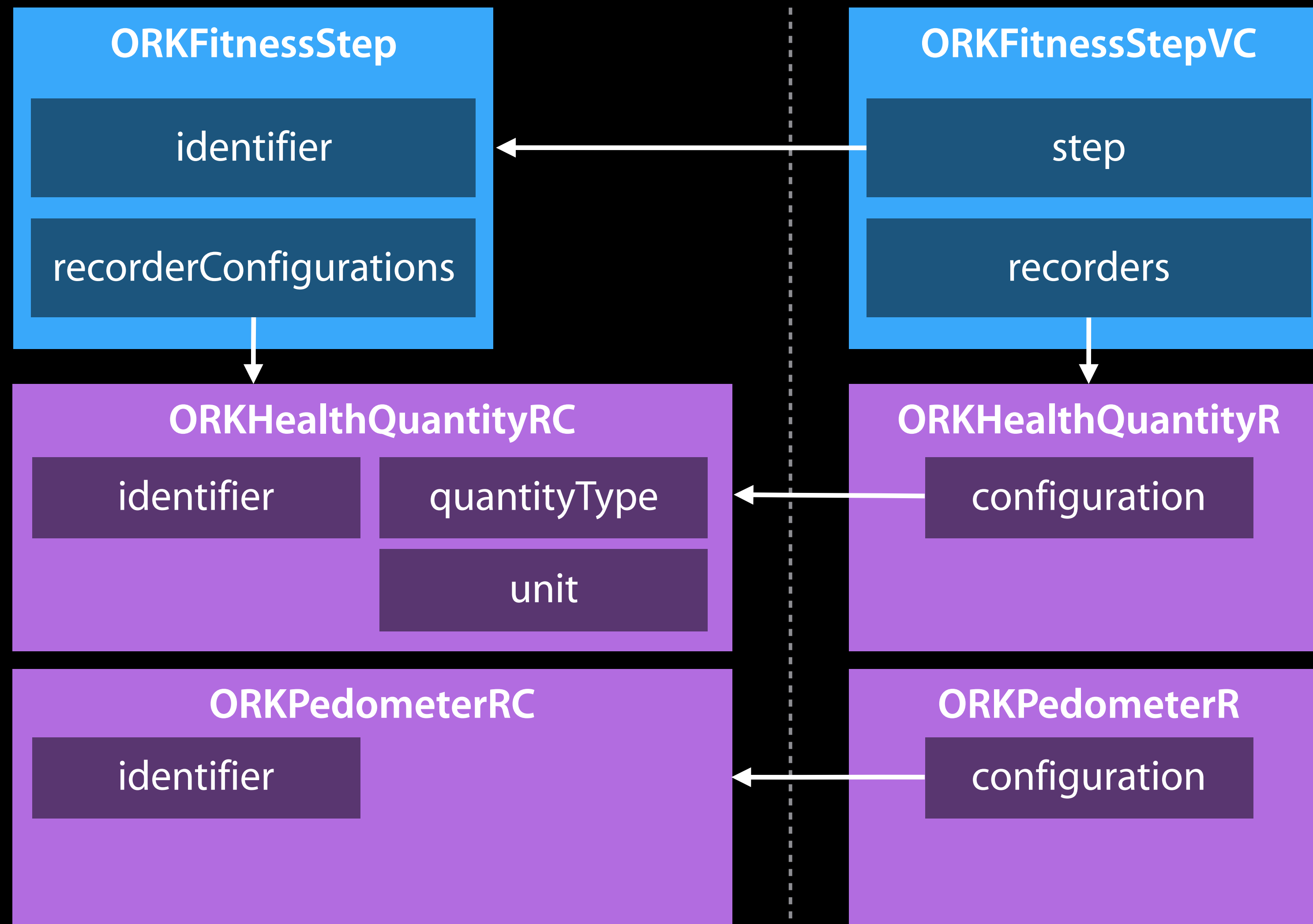
# Using Recorders



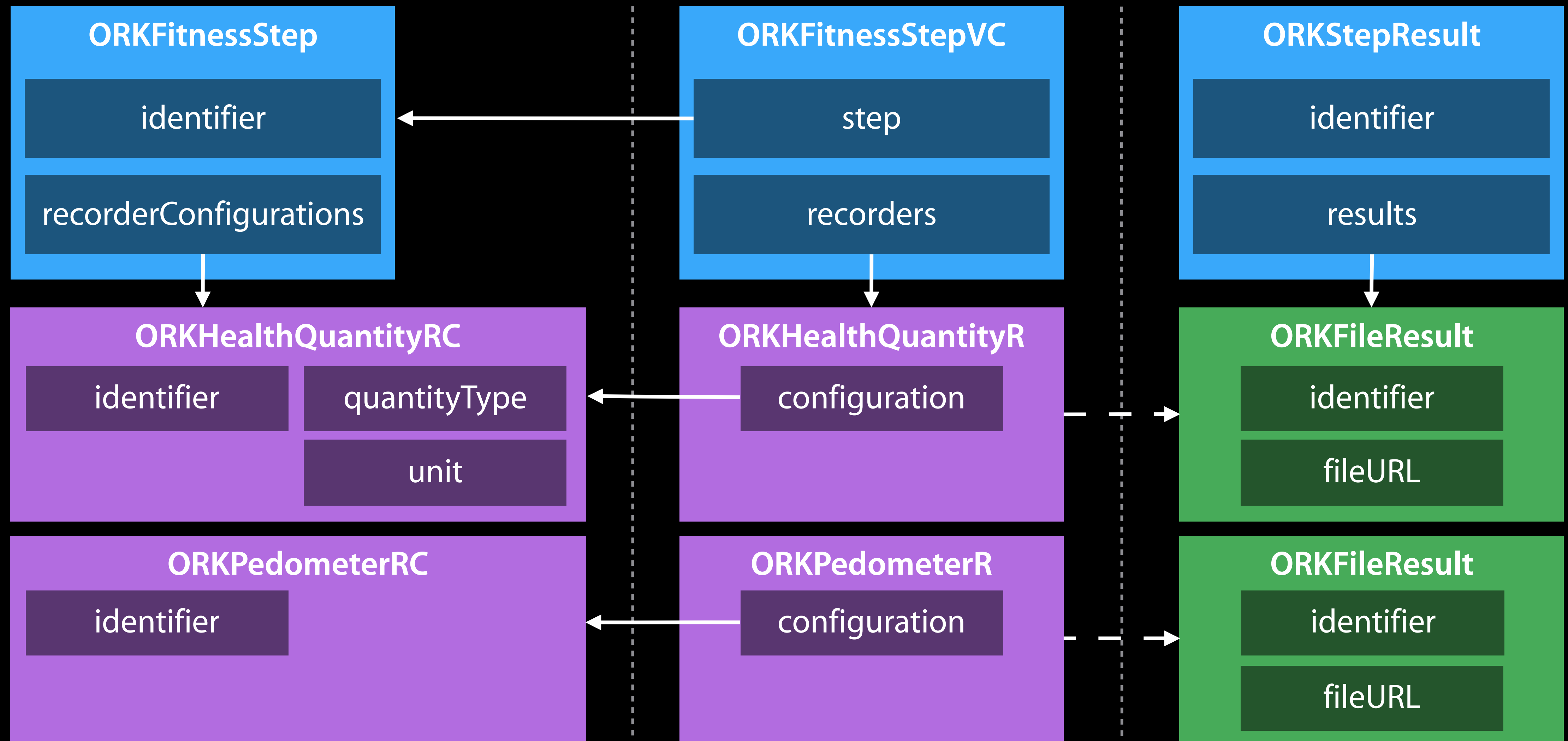
# Using Recorders



# Using Recorders



# Using Recorders



# Making a Custom Active Step

Subclass ORKActiveStep

Subclass ORKActiveStepViewController

Build UI

Configure recorders (if needed)

Add result classes (if needed)

# Summary

ResearchKit is open source  
Go and clone it  
Contribute!

# Summary

ResearchKit is open source  
Go and clone it  
Contribute!

It is an open project — will become what YOU make it

# More Information

## ResearchKit

ResearchKit Framework on GitHub  
[researchkit.org](https://researchkit.org)  
[github.com/researchkit/researchkit](https://github.com/researchkit/researchkit)

## Technical Support

[researchkit-users@lists.apple.com](mailto:researchkit-users@lists.apple.com)  
[researchkit-dev@lists.apple.com](mailto:researchkit-dev@lists.apple.com)

## General Inquiries

Contact for Principal Investigators  
[researchkit@apple.com](mailto:researchkit@apple.com)

# Related Sessions

What's New in HealthKit	Pacific Heights	Tuesday 11:00AM
HealthKit and ResearchKit Lab	Frameworks Lab C	Thursday 11:00AM
Health, Fitness and Research Get Together	Buena Vista Park	Wednesday 6:00PM

